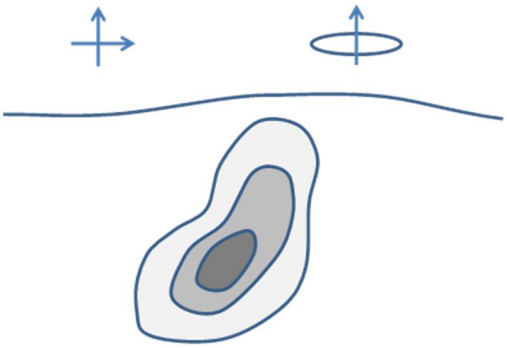
ArjunGUI

2D modelling and inversion of airborne time-domain EM data

User's guide to version 1.0



Markku Pirttijärvi

2014

University of Oulu

E-mail: markku.pirttijarvi(at)gmail.com

Contents

Contents....2
1. Introduction....4
1.1 Requirements and setup....5
2. User interface....6
2.1 Main GUI....6
2.2 Data GUI....8
2.2.2 Reading XYZ files....12
2.3 System GUI....14
2.4 Times GUI....17
2.5 Wave GUI....18
2.6 Model GUI....20
2.6.1 Model GUI widgets....20
2.6.2 Model GUI menus....25
2.7 Compute GUI....27
2.8 Results GUI....30
2.8.1 TEM sounding GUI....33
3. Program usage....35
3.1. Data input....35
3.2 Computational parameters....36
3.3. Model setup....37
3.4 Inversion and computation....39
3.4.1 Forward modelling....41
3.5 On the computational accuracy....41
4. File formats....42
4.1 Data files (AGD)....42
4.1.2 Generic data files....43
4.2 Geosoft XYZ files....43
4.3 System parameters files (AGS)....44
4.4 Time channel files (AGT)....45
4.5 Waveform files (AGW)....46

4.6 Model files (AGM).....47
4.7 Output mesh files (DAT).....48
4.8 GUI parameter file (DIS).....49
4.1 Sounding files (AGZ).....50
5. Additional information.....51
5.3 Terms of use and disclaimer.....52
References.....53

1. Introduction

ArjunGUI is a graphical user interface (GUI) for ArjunAir (Wilson et al., 2006), a computer program for modelling and interpretation of geophysical airborne electromagnetic (AEM) data from single profile using a two-dimensional (2D) model of electrical resistivity and susceptibility. ArjunAir was originally developed by Drs. Glenn Wilson, Art Raiche and Fred Sugeng for the CSIRO/Amira consortia (project P223F), and it became public domain software in 2010 (http://p223suite.sourceforge.net/). The modelling method is called 2.5 dimensional, because the model is two-dimensional and the (dipolar) source-receiver system is three-dimensional.

Presently, ArjunGUI supports only time-domain (transient) EM data and measurement systems. Because the focus is on airborne applications, the source loop is approximated as a magnetic dipole, i.e., the actual size and shape of the transmitter loop is not taken into account. The receiver is either a small loop or a coil measuring the time derivative of the magnetic flux density dB/dt (T/s) or a magnetometer (flux-gate or squid) measuring the intensity of the magnetic flux density B (T). The response is the vertical (Z) component or the horizontal (X) component along the flight line or both Z and X components.

ArjunGUI was developed primarily for inversion, but it does suit forward modelling too. In forward modeling the topography is flat and flight altitude is constant. In inverse modelling both the topography and the varying flight altitude can be taken into account. Inversion uses either unconstrained (SVD) or constrained inversion method to update the model resistivity values so that the data error, i.e., the difference between the measured and the computed data is minimized. In constrained (Occam) inversion the roughness of the model, i.e. the variation of the resistivity between neighboring points is minimized together with the data error. Occam inversion is directed towards a smoothly varying resistivity model if all the cells below the profile are free to change. The initial resistivity (and susceptibility) values and the fix/free status of the parameters can be edited manually to incorporate a priori data in the model. Weights can be assigned to individual data points or stations or channels to either include them in or exclude them out from the inversion.

1.1 Requirements and setup

The ArjunGUI program requires a PC with MS Windows 32/64-bit operating system and a graphics display with a resolution of at least 1280×1024. The amount of RAM is not as critical a factor as the CPU efficacy, because dynamic memory allocation is used and the computation becomes very slow using detailed models (> 5000 elements). The forward computation is based on finite element solution in frequency domain and the time domain response is obtained via Fourier (sine or cosine) transform.

The graphical user interface and visualization are based on the DISLIN graphics library (version 10.4) by Helmut Michels (http://www.dislin.de). ArjunGUI is built upon a main GUI view and separate GUI windows for: a) data input and selection of measurement lines, b) system parameters, c) time channels, d) waveform (pulse), e) model creation and editing, e) forward and inverse computations and e) visualization and output of the results.

To install the program, one needs to decompress the distribution file (ArjunGUI.zip) somewhere on the hard disk and a ArjunGUI folder will appear. ArjunGUI is a 32-bit application that calls for the actual forward and inverse computations either the 32-bit version (ArjunAir32.exe) or the 64-bit version (ArjunAir64.exe). The 64-bit version of ArjunAir uses multithreaded optimizations that require the presence of libiomp5md.dll, a dynamic link library for OpenMP parallelizations. The slower 32-bit bit version doesn't utilize parallel computations and should be used only in older computers. The distribution file also contains a short description file (_Readme.txt), this user's manual (ArjunGUI_manu.pdf), an example project file (Example.agp), data file (Example.agd), model file (Example.agm), and system, time and waveform files for the SkyTEM system (Skytem.ags, Skytem.agt and Skytem.agw).

2. User interface

ArjunGUI is normally started by "double clicking" its icon in Windows Explorer with computer mouse. Alternatively ArjunGUI can be started from within a command shell in which case the working directory should be the same as the program directory. When started, a console window appears showing information about the status of the program. If a shortcut is created on Windows desktop one should remember to verify that the start-in folder is the same as the program folder. Otherwise, additional work files may appear on the desktop.

2.1 Main GUI

At startup ArjunGUI reads parameters related to the user interface from ArjunGUI.dis file. If the file is not found, a new one with default parameter values is created automatically. The program then reads from the program directory the default project file (AGUI.AGP) which contains information about the system parameters, time channel and waveform used in the previous modelling experiment. It also contains the model and the path to the data file (*.AGD) used in the previous modelling. If the project file does not exist, default values are used for all parameters. If the path to the data file is lost, standard operating system "Open file" dialog appears and the user can browse the directories for the AGD file that contains the measured EM data or select a totally new data file. Although Geosoft XYZ files are supported, the data selected and used in modelling needs to be stored in ArjunGUI's own AGD format. See chapter 4 for the file formats.

The program then builds up the Main GUI window shown in Fig. 2.1. To perform modelling or inversion the user needs to take care of following prerequisites:

1) either measured data is read and properly defined or a profile is defined for synthetic forward modelling,
2) system, time channel and waveform parameters are defined, and
3) appropriate 2D model is defined.

The OK status of these prerequisites is identified by the corresponding check boxes at the right side of the Main GUI. The *DATA*, *SYSTEM*, *TIMES*, *WAVES* and *MODEL* buttons are used initiate separate GUI windows for modifications to data, system settings, time channels and waveform parameters and to the 2D resistivity (susceptibility) model.

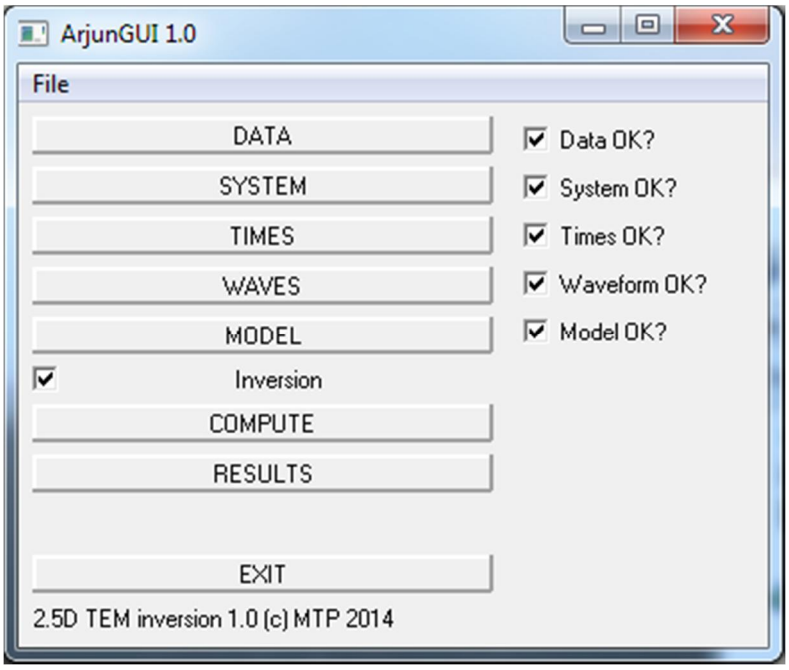


Figure 2.1 ArjunGUI Main GUI.

Computation can be started only if all the above-mentioned prerequisites are OK. Otherwise the *Inversion* check box and the *COMPUTE* button are inactive. If data has been read in the user needs to choose between inversion and forward computation using the *Inversion* check box before pressing the *COMPUTE* button. The *RESULTS* button is used to visualize the computed response and to compare the fit between computed and measured data.

The *File* menu of the Main GUI has only two items. *Open project* and *Save project* are used to open a previously saved AGP project file and to save the current modelling work into a project file, respectively. As discussed above, except for measured data all relevant information will be saved in the project file. Project can be saved only if the OK status of data, system parameters, time channels and waveform are all right.

On exit the program automatically saves the GUI parameters into ArjunGUI.dis file and the current status of the modelling work into AGUI.AGP project file. In an emergency case the user can press *Ctrl+C* keys in the console window, which prevents normal exit and saving of the project file.

2.2 Data GUI

Pressing the *DATA* button in the Main GUI brings up the Data GUI window shown in Fig. 2.2. It consists of a graph area on the right side and a vertical column of GUI widgets on the left side. The main purpose of the Data GUI is to:

1) visualize the TEM response as a function of profile distance,
2) to select and cut smaller anomalous parts of the profile data, and
3) to resample dense data and thus reduce the dimension of the linear problem to be solved.

If data are not defined, the graph area will be blank. Otherwise the upper graph shows the Z or X component of the TEM response where currently active time channel is shown by solid blue curve and other time channels are shown by dotted grey lines. The noise level, which should become visible at late time channels, is shown as a horizontal, green dashed line. The center time of the active time channel is shown above the plot. The index number of the active time channel is shown in the plot legend.

The lower graph shows the topography and flight line along the profile. The real X and Y coordinates of the beginning and end of the profile are shown between the two graphs. The vertical and horizontal axes of the lower graph are not in scale. In both graphs the horizontal axis is the profile distance, i.e., the distance from the beginning of the original (uncut) data profile.

Update data push button at the top of the widget column is used to validate changes made to the editable text fields (e.g. parameters of the synthetic profile). The *Invalidate data* push button is used to remove all information about the measured data and thus reverting program status from inversion to forward modelling.

Channel # text field and <-*Chan* and *Chan*-> buttons are used to identify and to change the active time channel. Pressing these buttons switches to the previous or next time channel, but it is also possible to jump into certain channel by first writing its channel number in *Channel #* field and then pressing either <-*Chan* or *Chan*-> button. *Swap component* button will to switch the data component shown in the graph between Z and X components.

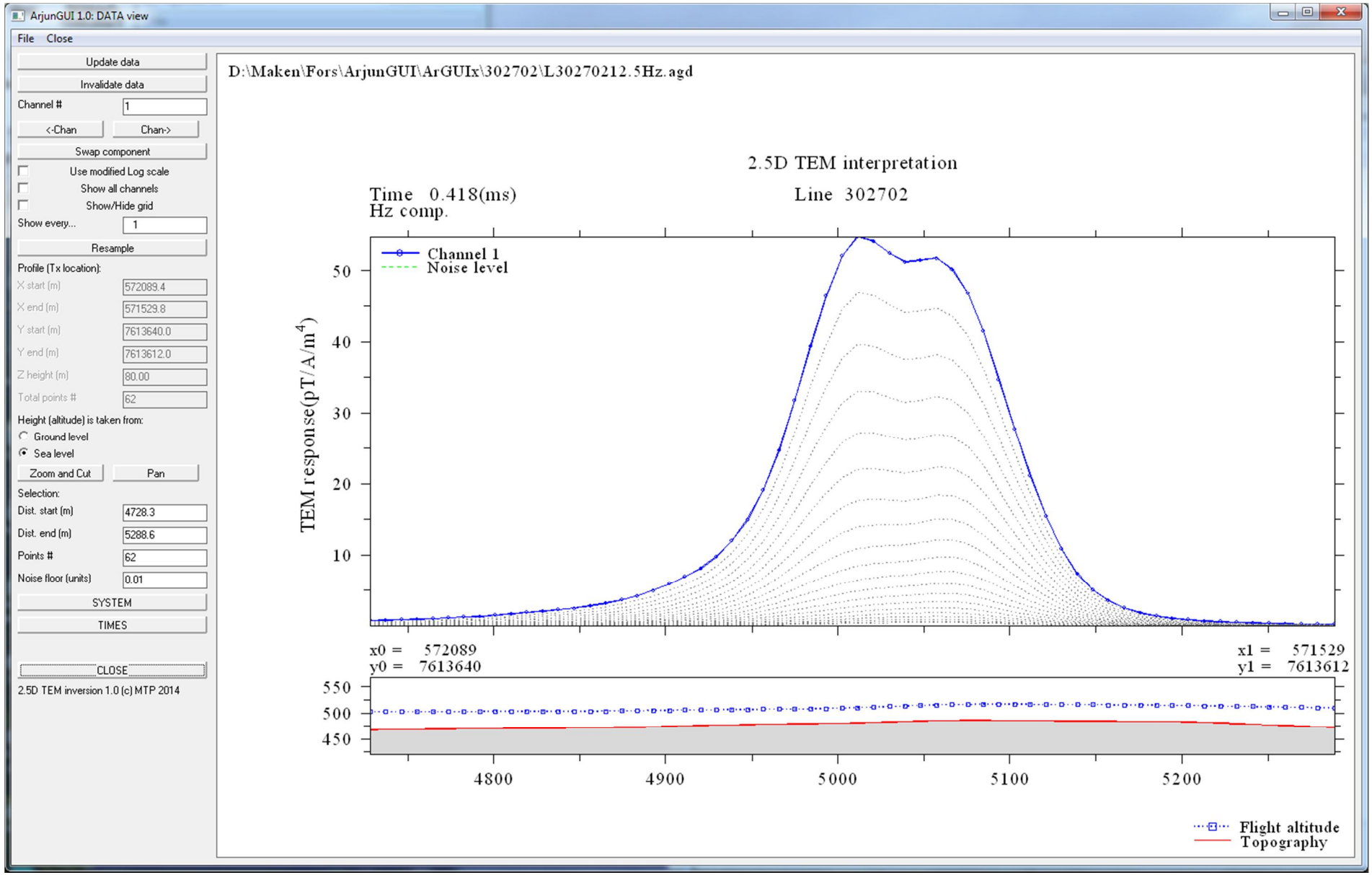


Figure 2.2 ArjunGUI Data view.

Use modified Log scale check box makes the graph y axis logarithmic. Because negative values cannot be shown on a logarithmic scale, a modified $\log_{10}$ axis is used, where the noise level (NL) (defined in System GUI) represents value 0. Data values greater than NL are shown above 0 (as if they were greater than 1), data values smaller than -NL are shown below 0 (as if they values between 0 and 1), and values between -NL and NL are not shown at all. *Show all channels* check box changes the way the minimum and maximum of the y axis are defined. Normally, they are based on the active channel. Alternatively, they are based on all time channels. *Show hide grid* check box is used to either show or hide the auxiliary dotted gray grid in the graphs.

Show every… text field defines data sampling. Quite often the original sampling of the TEM profile response is too dense and taking every *i*.th data point reduces the number of data and hence gives rise to smaller linear system to be solved in less time. The *Resample* button is used to update the value of *Show every…* field and to perform the actual resampling. The user should test different sampling values and choose the one that gives curves that are not too rugged but still show the details of the original data.

The six text fields under *Profile (Tx location)* label define the X and Y coordinates of the beginning and the end of the (original) data profile and the number of points on the original (un-resampled) profile. The text fields are inactive and cannot be edited when data has been read in. If data does not exist, they will be active and define the location of the synthetic profile.

Height (altitude) radio buttons define the flight altitude either as 1) elevation from ground level or 2) topographic elevation from sea level. The correct mode depends on the data and the appearance of the flight altitude curve in the lower graph can be used to verify the correct mode.

Zoom and Cut push button is one of the most important widgets of the Data view. It is used to select a single TEM anomaly from a long data profile. The selection is made manually or interactively using the computer mouse.

Manual zoom and cut is made by providing the profile start and end positions in the *Dist. start* and *Dist. end* text field and pressing the *Zoom and Cut* button. The start distance must be smaller than the end distance and there must be more than two points on the profile. The selection can be reset, i.e., the full profile can be made visible again by providing zeros for both the start and end distance (and pressing the *Zoom and Cut* button).

The interactive zoom and cut is made as follows:

Zooming and mouse editing mode 1: When Edit/Zoom mode is entered the cursor changes from an arrow into crosshair cursor and all the GUI widgets become inactive (to prevent other GUI tasks while in editing mode). By pressing the left mouse button over the response graph and keeping it pressed the user then draws a rectangular selection area. The outline (rubber-band) of the selection stays visible while the left mouse button is being pressed. After releasing the mouse the graph will be updated and shows only the selected part of the profile.

The program stays in the editing mode for new refined selection. Edit/Zoom mode is exited pressing the left mouse button once without moving the mouse. Edit mode 1 is used also when editing model resistivities in Model GUI.

Pan push button is used to re-center the zoomed area. The panning is made as follows:

Panning and mouse editing mode 2: When Edit/Pan mode is entered the cursor changes from an arrow into crosshair cursor and all the GUI widgets become inactive. The new center for the updated zoom view is selected by pressing the left mouse button once on the response graph. Pressing the right mouse once will cause the graph to be updated but the program stays in the editing mode and waits for new center.

Edit/Pan mode is exited pressing the right mouse button once without giving new center location with the left mouse button. Edit mode 2 is used also when painting model resistivities in Model GUI.

The three text fields below the *Selection* label text show the start and end profile distance and the number of points on the selected part of the profile. These values cannot be edited manually. *Noise floor* text field is used to change the value of noise level of the active TEM data component (Z or X). The noise level can be set also from within the System GUI, but it is included into Data GUI because here it can be compared to the actual (late time) data better.

Hint: Noise level is used in the inversion as a data validation threshold. All values whose absolute value is below the noise level are automatically removed from inversion, i.e., given zero weight.

SYSTEM and *TIMES* buttons are the same as in the Main GUI. In Data GUI they allow quick access to parameters such as data units or normalization without the need to close the Data GUI window.

Read TEM data and *Save TEM data* menu items will read an existing AGD file and save the selected and resampled data into an AGD file, respectively. The *Read XYZ file* item is discussed in the next chapter. The remaining items in the *File* menu allow saving the data graph in Postscript

(PS), Encapsulated Postscript (EPS), Portable document format (PDF), Portable network graphics (PNG) and Graphic image format (GIF) files. The PDF format usually gives the best results. The snapshot tool of the PDF reader gives a possibility to get a good quality bitmap image of the graphs.

The *Close* menu contains two items. The first one is used to restart the Data GUI using either normal or modified aspect ratio. Normal mode suits screens with 3:4 aspect ratio. In modified mode the program asks for a multiplier for the aspect ratio. A value less than one (default is 0.8) suits widescreen displays better. The second menu item will close the Model GUI and return the program operation back to Main or Results GUI.

2.2.2 Reading XYZ files

Selecting the *File/Read XYZ data* menu item in Data GUI brings up the Read XYZ GUI (Fig. 2.3). The GUI consists of a text area at the top, a vertical table with two columns on the right, a small graph showing the TEM response next to it, and several GUI widgets below the text area. The purpose of the Read XYZ GUI is to provide an easy way to select correct coordinate and data columns and to preview the TEM response of the profile lines before selecting one for further processing. Whether or not the graph shows the TEM response correctly depends on the correct choice of columns.

The text area shows the header of the selected XYZ file and the first three lines of the first profile. The header contains all the text lines before the first instance of LINE directive. The maximum amount of header lines is limited 50 and only the first 230 characters are shown. For more information about XYZ file, please, read chapter 4.2.

The first four pull-down list widgets (*X column*, *Y column*, *Z column* and *Alt. column*) below the header text are used to define the index numbers of the columns corresponding to X, Y, and Z coordinates and the altitude H. Index value zero (0) is used for missing information.

Header row list widget defines the row number of the header that contains (or might contain) the column labels. The labels are then separated from each other and shown in the first column (*Head*) in the table on the right. The other column (*Row*1) shows the contents of the first row of the first profile (the line below the first instance of the LINE directive). The table is used to identify the coordinate and data column index numbers.

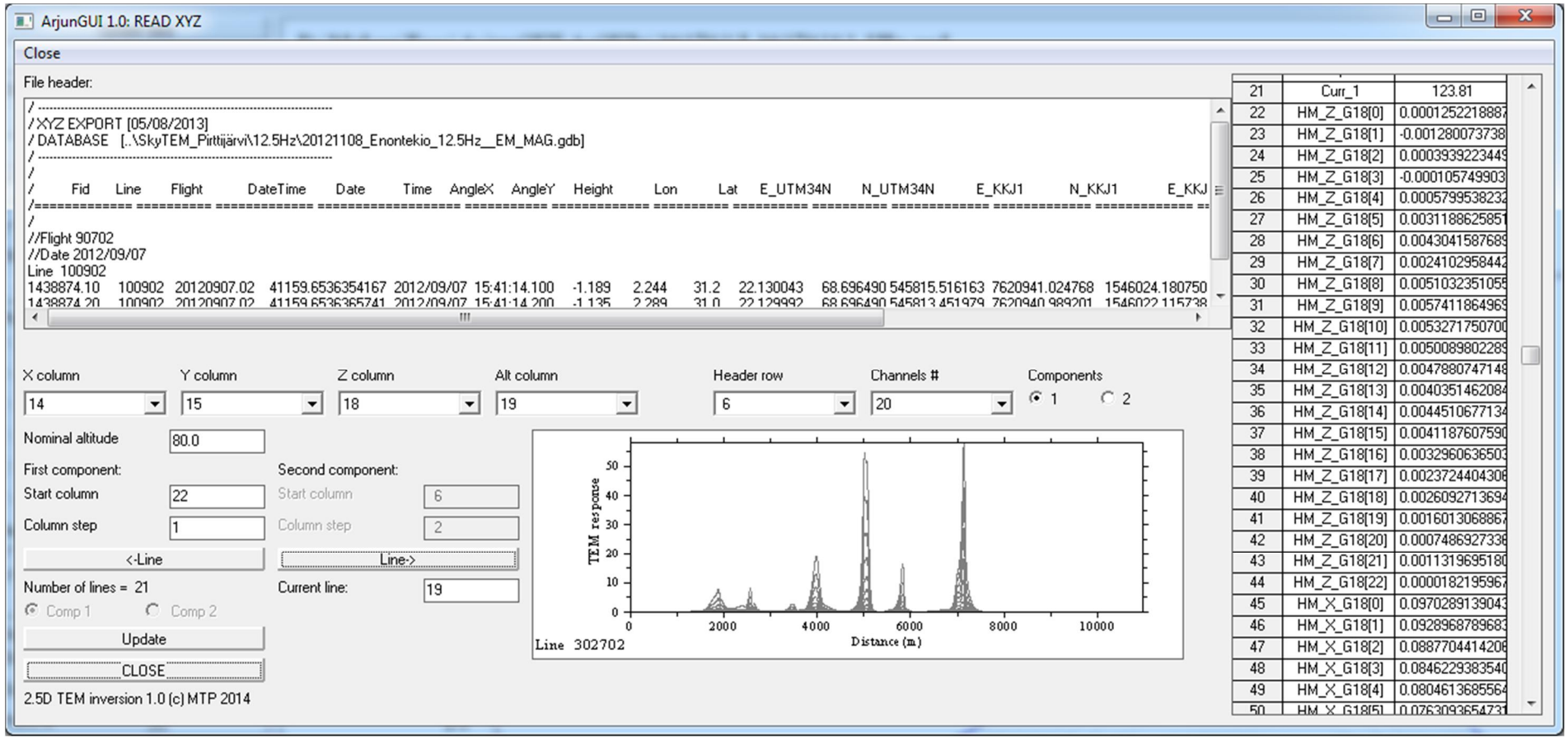


Figure 2.3 ArjunGUI Read XYZ view.

Channels # list widget defines the number of time channels (NCHAN) and *Components* radio button defines the number of response components (NCOMP) to be read from the file. *Nominal altitude* is used to define a constant value of flight altitude if that information is not read from the file. If the altitude is missing, the nominal altitude is added to the topography. If topography is missing it is assumed to be zero.

The *Start column* and *Column step* text fields are used to define the first column of the first and the second data component (active only if NCOMP= 2) and the column index step used to automatically define the column indices of the NCHAN time channels. This means that individual time channels must be evenly distributed (every *i.*th column) in the file.

Initially the graph will show the first profile line of the XYZ file. The *<-Line* and *Line->* buttons are used to change the active line backward and forward. The number of lines contained in the XYZ file and the index number of the active line are shown below the buttons. Note that the line title or name (defined in the LINE directive) is shown in the lower left corner of the graph. If two data components are to be read the *Comp 1* and *Comp 2* radio buttons can be used to change the active graph between Z and X component data.

Note: If two data components are to be read the first one (*Comp 1*) must refer to the Z component and the second (*Comp 1*) one must refer to the X component.

Update button must be pressed to validate changes made to the text fields (*Nominal altitude*, *Start column* and *Column step*) and to update the graph accordingly. The *Close* button closes the Read XYZ GUI and returns to Data GUI with the active profile line selected.

Hint: In order to quickly change from one profile to another the whole XYZ file is read into program memory before the Read XYZ GUI is build up. The user should be patient when opening large XYZ files because reading may take a while. The console window can be used to check if reading is still in progress.

2.3 System GUI

System GUI window is shown in Fig. 2.4. It consists of three columns of widgets. The *Update system* button is used to validate the changes and to refresh the status of the widgets.

The radio buttons at the top left corner define the domain mode (DMODE) for automatic or user defined TEM computations (DMODE= 1 or 2) and b) FEM (frequency-domain EM) computations (DMODE= 3). Presently, FEM computations are not possible in ArjunGUI.

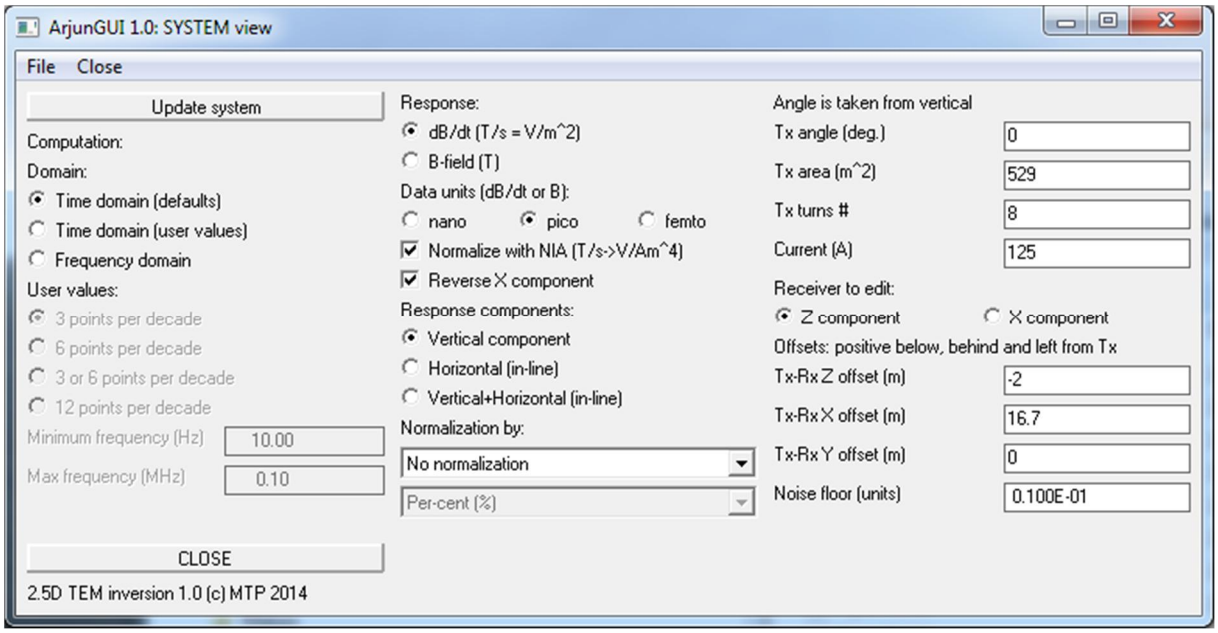


Figure 2.4 ArjunGUI System parameters view.

ArjunAir computes the frequency-domain response first over a wide frequency range (for 21 wavenumbers Ky along strike direction) and then applies Fourier transform for the time-domain response. When user defined domain mode is activated (DMODE= 2) the remaining widgets on the left column become active and define the points-per-decade (ppd) mode and the minimum and maximum frequency (MINF in Hz & MAXF in MHz) for the FEM range. The ppd modes are: 1) 3 ppd between MINF and MAXF, 2) 6 ppd between MINF and MAXF, 3) 3 ppd between MINF and 10 kHz and 6 ppd between 10 kHz and MAXF, and 4) 12 ppd between MINF and MAXF. In the automatic TEM computation mode (DMODE= 1) 3 ppd is used between MINF= 1 Hz and 10 Hz and 6 ppd is used between 10 Hz and MAXF= 100 kHz. The automatic mode is a good compromise between accuracy and computation time. On one hand, time could be saved particularly during inversion by limiting the frequency range. On the other hand, high maximum frequency and fine ppd is needed to model very early times accurately.

Response radio buttons at the top of the middle column define the response mode (RMODE) between time derivative of the magnetic flux density (dB/dt, T/s = V/m^2) and intensity of magnetic flux density (B-field, T). The *Data units* radio buttons are used to define the scale of

the measured (and computed) dB/dt data either as nano (10^{-9}), pico (10^{-12}) or femto (10^{-15}) T/s (or nT, pT, fT in case of B-field data).

Normalize with NIA and *Reverse X component* check boxes at the center of the System GUI are used to rescale the measured dB/dt data for the inversion and to reverse the sign of the X component data. Quite often measured data is normalized with the effective moment of the transmitter loop (NIA). By multiplying the data with the number of transmitter loop turns (N), the current (I) and the surface area of the loop (A), measured data defined as V/(Am^4) will become V/m^2 = T/s which is used in ArjunAir for dB/dt response. Reversal of X component sign might be useful in some cases.

Response components radio button is used to define which response components (Z or X component or both of them) are modelled or used in the inversion (CMODE). When modelling both components the first one is always the Z component and the second one is the horizontal X component which is equal to the in-line component along the flight line. The transverse Y component is not considered at all.

The two pull-down lists are used to define the response normalization mode (NMODE) and the units of normalized data (NUNIT). Response normalization, which is usually used only for FEM data, has following options: 1) un-normalized data, 2) response is normalized with vertical primary field, or 3) with horizontal primary field, 4) data is normalized with corresponding primary field component (Z component divided by vertical and X component with horizontal primary field) or 5) data is normalized with the total primary (dB/dt or B) field. The normalized data can then be defined in following units: 1) percent (%), 2) permille (‰), 3) per-million (ppm), or 4) for per-billion (ppb) meaning that the normalized data are multiplied with 100, 1000, 10^6 or 10^9, respectively.

The four text fields at the top of the right column define the transmitter: 1) the angle of the transmitter loop (degrees from vertical, positive if the front of the loop is above the rear of the loop), 2) the surface area (m^2) of the transmitter loop, 3) the number of turns in the transmitter loop, and 4) the transmitter current (A) used in data (NIA) normalization and when the waveform is given using normalized values of current.

The four text fields at the bottom of the right column define the Z, X and Y offsets (m) of the receiver with respect to the transmitter, and the noise level of the data components. The offset

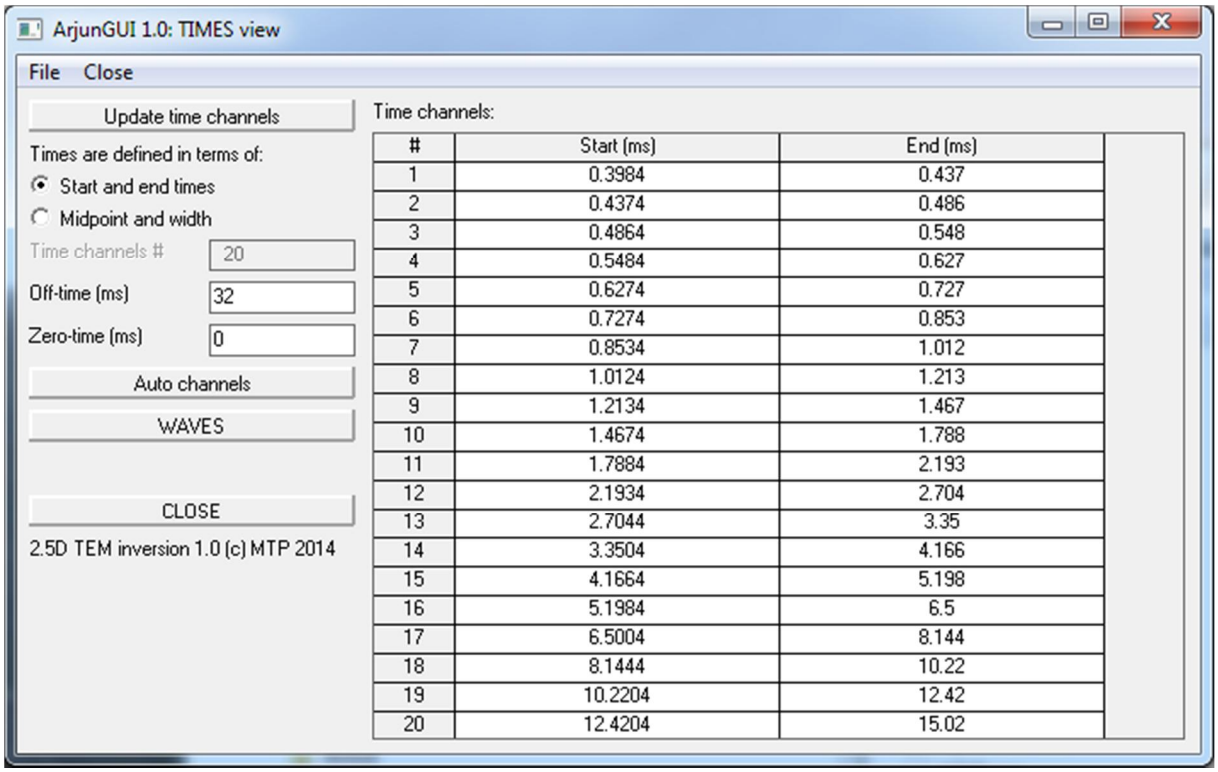
values are positive if the receiver is below, behind and left to the transmitter loop. The *Receiver to edit* radio buttons at the center of the right widget column are used to swap the receiver offsets and noise level between the Z and X component receiver.

The two menu items *Read system* and *Save system* are used to import and export system parameters from and into an AGS file.

Before closing the System GUI user must press the *Update system* button to validate any changes made to the text values of the system parameters.

2.4 Times GUI

The Times GUI (Fig. 2.5) consists of a column of GUI widgets on the left side and a table with two editable columns on the right side. The *Update time channels* button is used to validate the changes made to the text fields or table.



#	Start (ms)	End (ms)
1	0.3984	0.437
2	0.4374	0.486
3	0.4864	0.548
4	0.5484	0.627
5	0.6274	0.727
6	0.7274	0.853
7	0.8534	1.012
8	1.0124	1.213
9	1.2134	1.467
10	1.4674	1.788
11	1.7884	2.193
12	2.1934	2.704
13	2.7044	3.35
14	3.3504	4.166
15	4.1664	5.198
16	5.1984	6.5
17	6.5004	8.144
18	8.1444	10.22
19	10.2204	12.42
20	12.4204	15.02

Figure 2.5 ArjunGUI Time channels view

Time channels can be defined in two different ways: a) as start and end times or b) as the midpoint and width of the time channel. Pressing the corresponding radio buttons will change the table on the right accordingly. All time values are defined in milliseconds (ms).

Time channels # text field defines the number of time channels (NCHAN) and, hence, the number of rows in the adjacent table. If data has been read in the number of time channels is fixed and the corresponding text field will be inactive.

Off-time is the length (in ms) of the time period between the zero-time and the start of the next current pulse (with reverse polarity). In most TEM systems the response is measured during off-time. *Zero-time* defines the moment (in ms) on the waveform where the time channels are counted from.

The *Auto channels* button is active only in modelling (when data has not been read in). When it is pressed the program asks for minimum and maximum times and automatically computes NCHAN time channels that are distributed evenly on a log-scale.

The two menu items *Read times* and *Save times* are used to import and export time channels from and into an AGT file.

2.5 Wave GUI

The Wave GUI view (Fig. 2.6) consists of a column of GUI widgets on the left side, a table with two editable columns on at top of the graph, and a graph which shows the distribution of time channels on a log-scale in the upper panel and the current pulse and time channels on the lower channel. The *Update waveform* button is used to validate the changes made to the text fields or table.

The waveform can be defined two ways: 1) values of current in the source loop or 2) values of calibration data measured during calibration flight. Both are given as digitized curves as a function of time (WMODE= 1 or 2) and the behavior of the current (or calibration data) is assumed to vary linearly between the time nodes. The time values are given in milliseconds (ms) and current values are in amperes (A). If calibration data is used, the user needs to define whether the data are 1) in-line dB/dt, 2) in-line B-field, 3) vertical dB/dt, or 4) vertical B-field

(XMODE= 1, 2, 3, or 4). In this case the units of the calibration data are those of the measured data (nT/s, pT/s, fT/s or nT, pT, fT).

Off-time and *Zero-time* text fields are the same as in Time GUI. Zero-time allows defining the time abscissae of the waveform independent from the actual time channel values. Note that usually the zero-time is equal to the moment where the pulse starts to decay (as in Fig. 2.6) and not the end (or the beginning) of the waveform. Thus, if the time abscissae of the waveform were to start from zero in Fig. 2.6, then the zero-time would need to be 8.0 ms to give the same result.

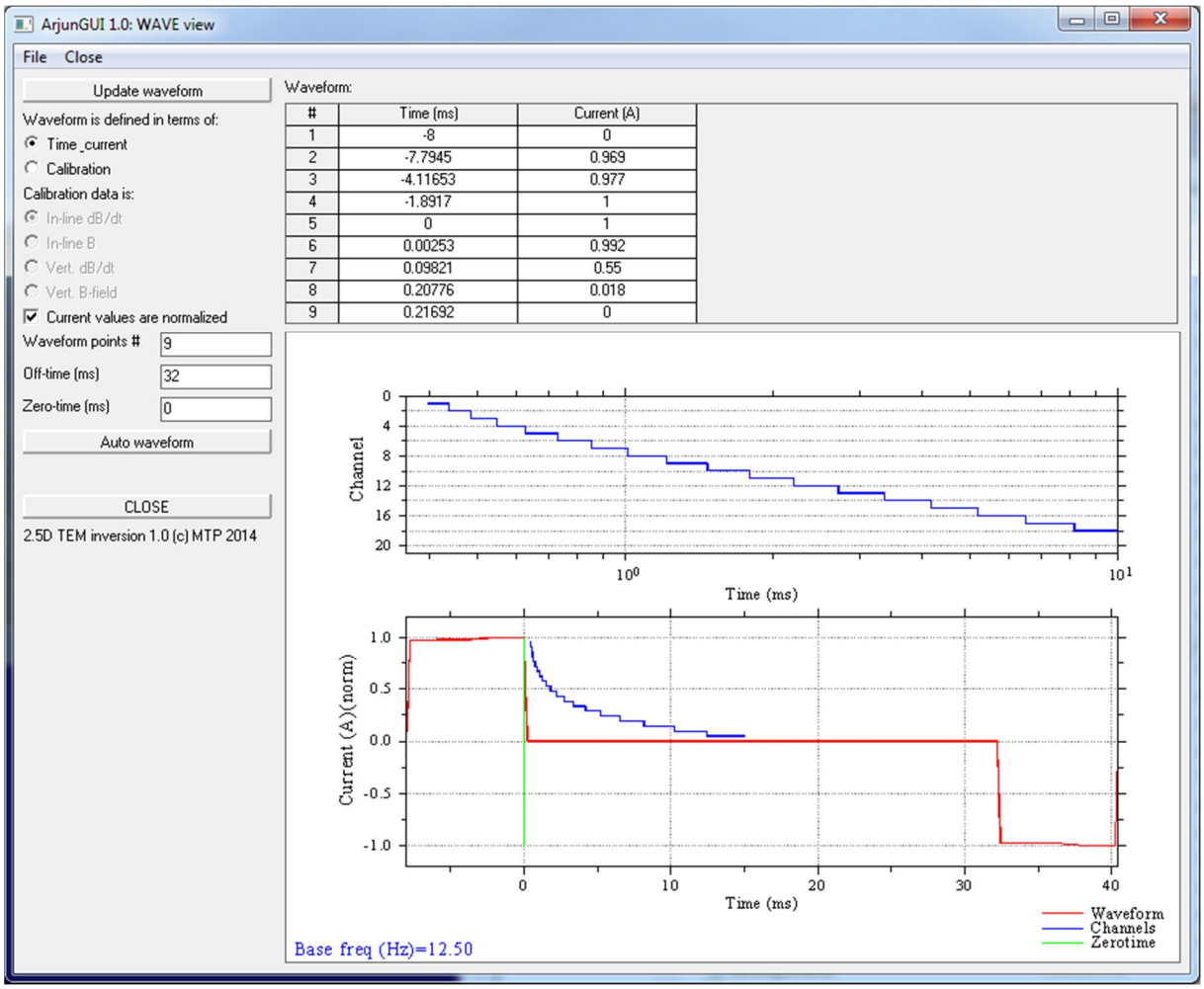


Figure 2.6 ArjunGUI Waveform view

The current values can be absolute or relative which means that they have been normalized with the peak current given in the System GUI. If the check box *Current values are normalized* were to be unchecked in Fig. 2.6, then the values of source current in the table would be interpreted as absolute values in amperes.

Auto waveform button is active only in modelling (when data has not been read in). When pressed the program asks for start and end times of the pulse, the maximum current and the length of a current ramp before automatically computing a simple four-point current pulse.

The graph, which shows the pulse shape and the distribution of time channels, can be used to find out problems in the waveform and time channels. The upper graph is shifted according to the zero-time, and hence, on-time channels will not be shown on the logarithmic scale. The lower graph uses linear scaling, and therefore, can be used to compare time channels and waveform better. The base frequency of the TEM system is shown in the lower left corner of the graph. The base frequency (in Hz) is based on the width of the off-time and waveform as: *Base_freq*= 1/(2*(*off-time* + *wave-time*)), where wave-time is the length of the waveform before the zero-time.

The two menu items *Read waveform* and *Save waveform* are used to import and export waveform from and into an AGW file.

2.6 Model GUI

The Model GUI (Fig. 2.7) is perhaps the most advanced GUI window of ArjunGUI application. It consists of a single column of GUI widgets on the left and a large graph area on the right side of the window. The graph area will be blank (and model parameter text fields will be inactive) if model has not yet been defined. Otherwise the graph shows a vertical cross-section of the 2D resistivity (or susceptibility) model. The colors represent the 10-base logarithm of the resistivity (or susceptibility) variation. By default, the TEM data profile and its measurement points are shown by small open circles above the model graph. The graph title consists of the text string read from ArjunGUI.dis file and the data header (AGD) or line header (XYZ). The information text on the left bottom corner of the graph shows: a) geometric scale between vertical Z and horizontal X axis of the graph, b) the amount of fixed and free cells, c) the total amount of model nodes (NNODE), and d) the model discretization, that is the number of cells in horizontal and vertical directions (INODE, JNODE).

2.6.1 Model GUI widgets

Generate initial model button is used either 1) to create a totally new initial model based on the measured data or synthetic profile defined in Data GUI or 2) to update the current model

with new model parameter values (discussed later). Pressing the button will not only change the model position and discretization but also resets any existing resistivity (and susceptibility) back to its initial value. For susceptibility the initial value is k = 0 SI. For resistivity the initial value is the user given background value in case of forward modelling. In case of inversion, i.e., when data has been read in, a special transformation method is used to compute initial resistivity distribution based on the TEM data. *Discard model* button is used to remove all information on current model.

The initial model consists of quasi-rectangular elements or cells defined by the four neighboring nodes in the corners. The total number of model cells is NCELL = (INODE-1)*(JNODE-1). The beginning and the end of the model are based on the start and end of the data profile (in both forward and inverse modelling). The nominal width (DX) and nominal height (DZ=DX/2) of the cells is based directly on the mean data sampling. The size of the elements (DX×DZ) should depend on the resistivity of the model. The more conductive the model is the smaller the size of the elements should be.

The width of the elements varies along the profile so that for the middle of the profile, where the anomalous conductivity structure should locate, has the nominal width (DX), the start and end of the profile have twice as big cell width (2×DX) and the margin areas added before the start and after the end of the data profile have cell width equal to 4×DX. The margins are needed to better model the decaying behavior of the EM fields. The width of the margins is defined by user and normally they should be more than 250 m wide.

In the initial model, the height of the cells increases downwards by a factor of one. Thus, if DZ=DX/2 is the height of the top-most element the height of the elements increases as: DZ, 2×DZ, 3×DZ, 4×DZ, etc. The topography along the profile is taken into account and the height of the elements varies in such a way that the bottom of the model becomes horizontally flat. The depth to the bottom of the model is a user defined parameter. Normally the total height of the model should be more than 500 m. The actual depth to the bottom depends on the vertical discretization of the model because it must be an even number of DZ.

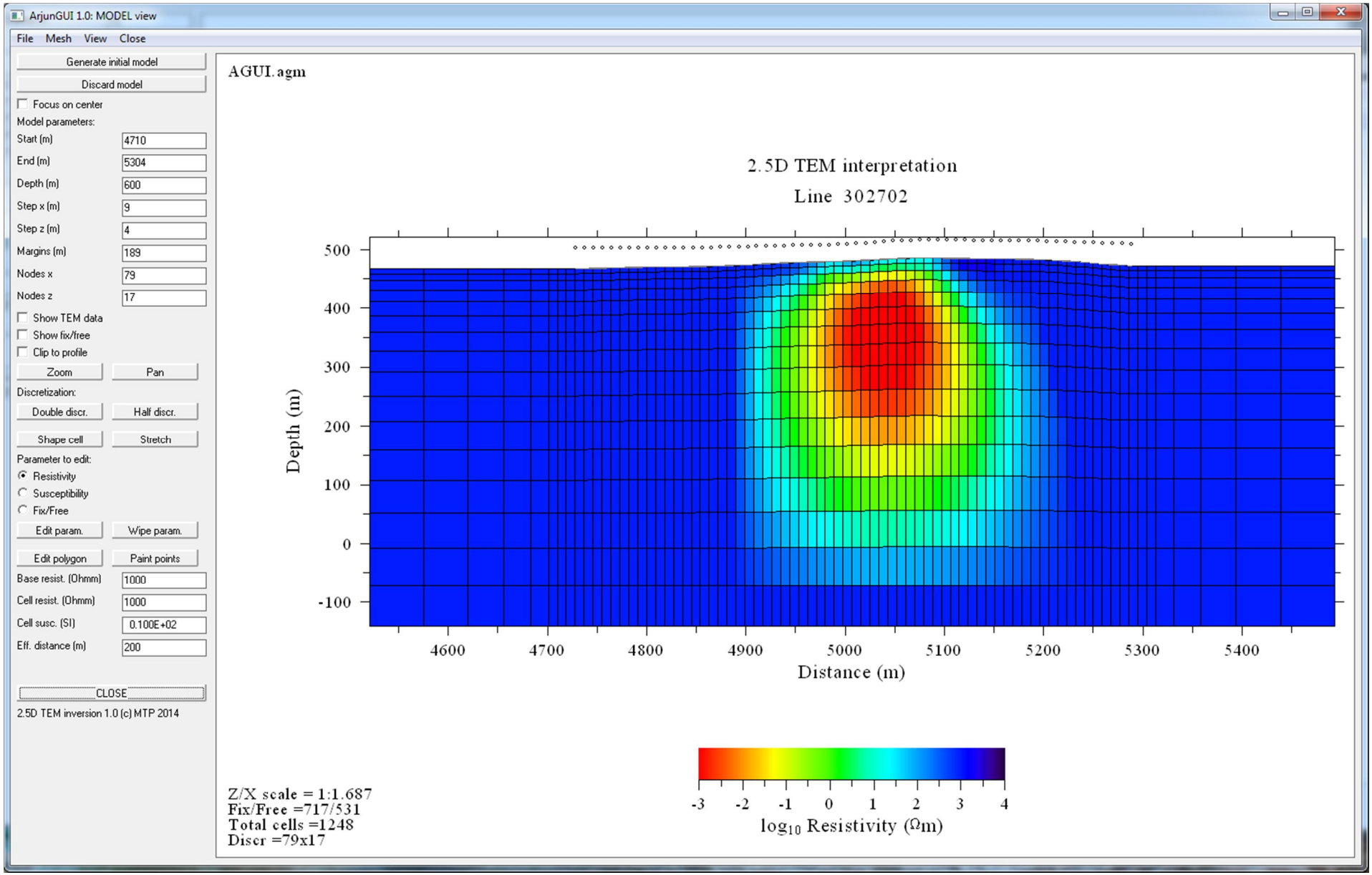


Figure 2.7 ArjunGUI Model view.

When an initial model has been generated the eight text fields below *Model parameters* label define: a) the start and end profile location of the model in profile coordinates (from the beginning of the original, un-cut data profile), b) the (nominal) height of the model, c) the (nominal) width and height of the cells, d) the width of margin areas before the start and end of the profile and e) the number of model mesh nodes in horizontal and vertical direction.

The three check boxes *Show TEM data*, *Show fix/free* and *Clip to profile* are used to change the graph appearance. TEM data profile can be shown below the graph in place of the color scale. Fixed cells can be identified so that a cross is drawn over them. Margin areas can be clipped away from the model view so that only the part of the model below the profile remains visible. When clipping is active the vertical depth axis will be shown in scale with the horizontal x axis, unless zooming is also active.

Zoom and *Pan* buttons behave like those in Data GUI (see chapter 2.2). They allow more precise editing of the model nodes.

Double discr. and *Half discr.* buttons activate an interactive editing mode (Edit mode 1, in chapter 2.2) where the discretization of the cells under the selection rectangle are either doubled or halved. The operation stays in editing mode until a null-selection is made (left mouse button is pressed without moving the mouse). The operations are made in both x and y direction and they extend across the whole model. To reduce discretization along x or y direction alone, the user should draw a thin rectangle that stays inside a single row or column of elements. If the discretization is already very dense, the user is should zoom in for better view for the model.

Shape cell and *Stretch cells* buttons activate an interactive editing mode (Edit mode 1) where the nodes, i.e. the corners of the elements can be shifted with mouse. *Shape cell* affects only the nearest node and moves the corner by a maximum of 75% of the size of the cell (neighboring cells are not affected). *Stretch cells* operation affects several nodes at the same time extending the shift operation over an area the radius of which is defined in *Eff. distance* text field below. The operation stays in editing mode until a null-selection is made (left mouse button is pressed without moving the mouse).

Parameter to edit radio buttons are used to define the parameter to show in the graph (resistivity or susceptibility) on one hand and the parameter (resistivity, susceptibility or fix/free status) to edit on the other hand.

Edit param. and *Wipe param.* buttons activate interactive editing mode (Edit mode 1) where (depending on the status of the *Parameter to edit* radio buttons) the resistivity or susceptibility or the fix/free status of the model can be edited with the mouse. As discussed in chapter 2.2, the editing is done by drawing a rectangle with left mouse button. The operation stays in editing mode until a null-rectangle is made (left mouse button is pressed without moving the mouse).

Hint: Between each new selection, the parameter value (base resistivity, cell resistivity or susceptibility) is read from the corresponding text fields below the four edit buttons so it is possible to change the parameter value while working inside the editing mode, i.e., between each separate rectangle.

Edit polygon button activates an interactive editing mode (Edit mode 2) that allows editing of more polygonal shapes. Polygons are defined pressing the left mouse button multiple times at the vertices of the polygon. The shape of the polygon will not be shown on the screen so the user must select the points carefully. The polygon is closed and the model is redrawn after pressing the right mouse button once. The operation stays in polygon editing mode until user gives a null-polygon (right mouse button is pressed without defining any polygon vertices). Note that polygon mode does not allow wiping the fix status of the cells – one needs to use the rectangular wipe mode for that.

Paint points button activates yet another editing mode (Edit mode 2) which allows editing of smooth resistivity and susceptibility distributions. The idea is that the user insert points with given parameter value (*Cell resist.* or *Cell susc.*) pressing the left mouse button once or multiple times over the model and then ends the insertion pressing the right mouse button once. The existing parameter values of the cell below and radially around the given points are affected by inserted point(s) and the model is updated accordingly. The new value is a weighted mean of the existing and new values. An inverse distance weighting method based on the value of *Eff. distance* text field is used to set the effective distance of the inserted point. According to the formula $1/10^{(2*r/r_{eff})}$ the weight is one below the insertion point (r=0)

and 0.01 at distance $r = r_{eff}$. Thus, the effective distance should be larger than the cell size to see any difference in the painted model. ON the other hand effective distance cannot be greater than half of the height of the whole model. The editing mode is exited pressing the right mouse button once without any new points. *Paint points* editing mode does not work with fix/free status, because it is effectively a binary value (1= fixed, 0= free).

Note: Fixed cells are handled differently depending on the status of *Show fix/free* radio button. If fixed cells are shown then their values are preserved and not affected by editing and wiping. If fixed cells are not shown then all the cells of the model are free to change in the edit/wipe operations.

2.6.2 Model GUI menus

File menu contains items for file input and output. The *Read model* and *Save model* items are used to read existing model from a model file (*.AGM) and to save the present model into a AGM model file, respectively.

Read RES file and *Save RES file* can be used to read and to save resistivity and susceptibility values from or to a *.RES file. ArjunAir.res file is normally used to pass the model resistivity values to ArjunAir computation. Reading a RES file could be used, for example, to revert the model back to the previously computed one (provided that the model geometry has remained the same). Saving RES files can be used for backup purposes and to pass the model information to the (modified version of) AMIRA's Arjuna program, which suits modelling borehole data and coincident loop data and where the geometry and size of the transmitter loop is taken into account.

Save mesh DAT is used to save the model resistivity and susceptibility values (and their 10-base logarithm) and fix/free status into a simple column-formatted text file to be used to visualize the resistivity sections in third-party software (eg. GoCAD). The remaining menu items in the *File* menu allow saving the model graph in Postscript (PS), Encapsulated Postscript (EPS), Portable document format (PDF), Portable network graphics (PNG) and Graphic image format (GIF) files.

Mesh menu contains items for handling some mesh related operations. The *Shift X node coordinates* moves the whole mesh forward or backward along the profile by an amount that

is less than the nominal width of the cells. The *Shift cell contents* shifts the cell parameter of the whole model to their neighboring cell on the left, right, below or above.

Multiply values item multiplies all the resistivity (or susceptibility) values by a given value. Giving the inverse of the value (eg. 1/1000 = 0.001) means that the cell resistivities are divided by that value. The multiplication can be performed for the whole model or only for those values that are different from the background value.

Define overburden allows creating an overburden layer of fixed resistivity and thickness to cover the whole model. When initiated the program first asks for the vertical number of layer cells and the resistivity to be used for the overburden. Then it will ask for the thickness of the overburden and automatically stretches the height of the cells so that a layer with given thickness and number of elements becomes possible. The height of the cells belonging to the overburden is made equal and the height of the cells below are adjusted accordingly.

Reset resistivity and *Reset susceptibility* reset the model resistivity or susceptibility back to its background value. Unlike background resistivity, which is a user defined parameter the background susceptibility is always equal to zero. This means that the background magnetic permeability is equal to the free-space value ($\mu=\mu_0= 4\pi 10^{-7}$ As/Vm). Since zero values cannot be shown on logarithmic scale background susceptibility values are shown as blank (white) cells.

Add/delete columns/layers can be used to a) add (or delete) a layer to (or from) the bottom of the model and b) to add (or delete) a column to (or from) both sides of the model.

View menu contains items for that affect the appearance of the model graph. Depending on the status of the *Parameter to edit* radio button, the *Set res/sus scale* gives user on option to change minimum and maximum values for the resistivity and susceptibility. Note that in inversion these are used as absolute limit values for resistivity.

Solid/Dotted/No grid changes the way the lines between the mesh cells are drawn or hides the lines altogether. Similarly, *Show/Hide flight line* item can be used to hide the small circles representing the fligh line above the model. *Show/Hide labels* item is used to display small text labels defining the actual resistivity (or susceptibility) values over the model mesh. The density of the labels is based on the step value the program asks when *Show/Hide labels* item

is utilized. Step value 3, for example, shows the label for every third cell. Step value 0 resets the display of labels and will hide them all.

Color scale submenu can be used to change the color scheme between normal and reverse grayscale, normal and reverse rainbow scale and so called temperature scale with shades of red color.

Close menu contains two items. The first one is used to restart the Model GUI using either normal or modified aspect ratio. In normal mode 3:4 screen aspect ratio is used whereas in modified mode the program asks for screen ratio. A value less than one (default is 0.8) suits widescreen displays better because otherwise the GUI window would extend outside the right side of the screen. The second menu item will close the Model GUI and return the program operation back to Main GUI or Results GUI.

2.7 Compute GUI

The Compute GUI (Fig. 2.7) consists of a single column of GUI widgets. Depending on the status of the *Inversion* check box in the Main (or Results) GUI, the push button at the top of Compute view reads either *START COMPUTATION* or *START INVERSION*.

In the first case (modelling) only three widgets are active; *Use previous results, Use ArjunAir32.exe,* and *Base level trend* check boxes. Use of the previous results allows performing the forward computation in a fraction of the time required for the full problem. This is accomplished by using the ArjunAir.frq file that contains the full frequency-domain solution at receiver locations and is automatically saved during the last ArjunAir computation. Thus, the most time consuming part of the TEM computation can be avoided provided that the model, time and waveform parameters have not been changed.

Use ArjunAir32.exe will use the 32-bit version of ArjunAir instead of the 64-bit version. The 64-bit version works only under 64-bit Windows and uses Libiomp5md.dll dynamic link library for OpenMP parallelizations. *Subtract base level* check box, which is discussed later, is active only if measured data has been read in.

If inversion was enabled the user first needs choose *Normalization type*, *Inversion method* and *Conversion criterion*. The *Survey S-norm* is useful at the beginning of the inversion because

the amplitude changes are preserved along the profile. The *Point P-norm* should be used with care because the sensitivity of different parts of the profile will not comparable anymore.

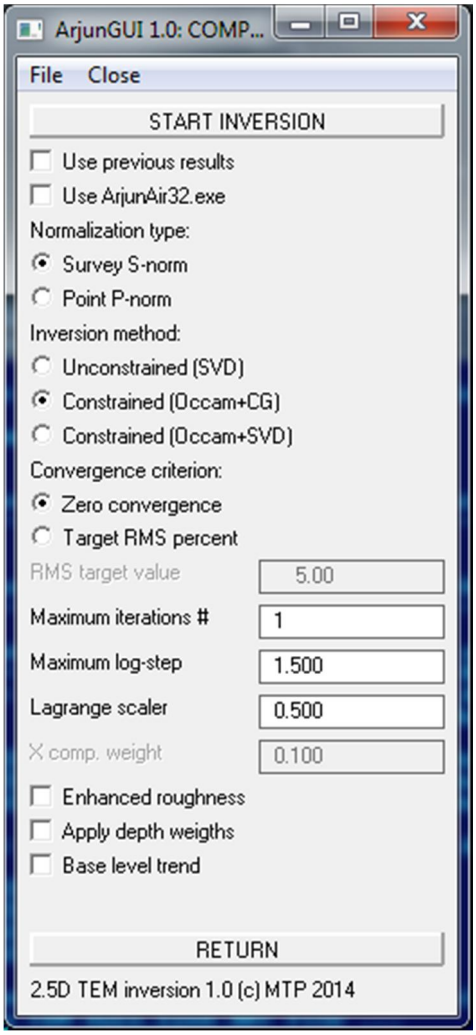


Figure 2.8 ArjunGUI Computation/Inversion view.

Unconstrained (SVD) inversion method should be used only when a model with a rather good fit already exists and few additional iterations are needed to find out as good fit as possible. Alternatively it can be used when an initial model exists and fix/free status of the cells have been edited so that the inversion will optimize the resistivity of the elements of the body or its neighboring cells only. C*onstrained (Occam+CG)* inversion is the default inversion method which produces rather smooth models because it aims to minimize the model roughness together with data misfit. It uses iterative conjugate gradient (CG) solver for the linear equations. The *Constrained (Occam+SVD)* inversion gives the benefits of Occam's method and the stable SVD method in solving the linear equation. It can be quite slow if the number

of free cells is large but it produces good fit and models that are not as smooth as in CG based inversion. *Constrained (Occam+SVD)* inversion should be used when making few final iterations on a model found out using C*onstrained (Occam+CG)* inversion.

The convergence criterion is either *Zero convergence* or *Target RMS percent*. In the first case, the inversion either performs the given number of inverse iterations (*Maximum iterations #*) or ends the iterative process when RMS becomes less than 0.1 %. In the latter case, *RMS target value* becomes active and inversion continues until either the maximum number of iterations or the target RMS value is reached.

Maximum log-step defines how much the resistivity values can change in single iteration step. It can be used as a stabilizing parameter especially at the beginning of the inversion. The maximum step is given on a log-10 scale so that the default value 1.5, for example, means maximum step of 1.5 decades (e.g. resistivity change from 10000 Ωm to about 300 Ωm).

Lagrange scaler is a parameter that is used as a measure between the model smoothness and data fit. The higher the value (1-10) is, the smoother the model will be and the worse the data fit will (usually) be. The lower the value (0.1-1) is, the better is the fit between measured and computed data and the rougher the model will appear.

X comp. weight gives a relative weight between the X and Z components in the inversion. Weight value 1.0 gives both data components equal importance. The default value 0.5 reduces the importance of X component with regard to the Z component, because often the X component data of real 3D structures does not fit well with a 2D modelling and sometimes simultaneous inversion of both Z and X component data is impossible. This item has no meaning when single component data is being used.

The three checkboxes, *Enhance roughness*, *Apply depth weights* and *Base level trend*, are used in inversion to a) produce a slightly more rugged and focused resistivity model, b) increase the sensitivity of the model elements at greater depths and c) define each time channel a large-scale linear trend based on the start and end points of measured data. The base level trend is then added to the computed data, and hence, can be used to define large-scale data variation that would otherwise need to be defined by the mesh elements outside the profile. These three items are more-or-less experimental and should be used with care.

After pressing the *START COMPUTATION* button, ArjunGUI generates ArjunAir.cfl control file and ArjunAir.res resistivity file, and calls for the ArjunAir**. program for the forward computation. Before computation ArjunGUI also generates a backup copy of the project file (AGUI.AGP) in the current working folder. During the computation ArjunGUI will be unresponsive. The progress of the computation can be followed from the console window. ArjunAir computes full frequency-domain spectrum for the 21 values of wavenumber Ky in strike direction. After the computation ends, ArjunGUI reads the results from ArjunAir.mf1 and closes the Compute GUI window.

After pressing the *START INVERSION* button ArjunGUI generates ArjunAir.cfl, ArjunAir.res and AGUI.AGP files (like in forward computation) and ArjunAir.inv inversion parameter file before calling the ArjunAir**.exe for the inverse computation. Because the sensitivity matrix (Jacobian) is computed during the forward computation, the inversion itself is pretty fast and most of the time is spend in the forward computations. If previous results are not used ArjunAir makes two forward computations, one to determine the initial computed response and the Jacobian and the other one to compute the response for the updated model (and the Jacobian). After the inversion ends, ArjunGUI reads the results from ArjunAir.mv1 and ArjunAir.res files and closes the Compute GUI. The ArjunAir.err file is used to check if errors occurred during the forward computation. ArjunAir.log and ArjunAir.out files are not used by ArjunGUI but they can be used as a source of additional information if needed.

File menu contains *Read CFL file* and *Save CFL file* menu items to handle ArjunAir control files. Presently, reading control files is not possible but saving them allows storing all the necessary information required to re-compute the current modelling problem. Please, note that to use the control files with ArjunAir.exe requires that the files are renamed as ArjunAir.cfl.

2.8 Results GUI

The Results GUI shown in Fig. 2.8 consists of a graph area on the right and a column of GUI widgets on the left. The graph shows the computed response so that the active time channel is drawn as a solid blue curve and other time channels are drawn as dotted gray lines. If data was read in the measured TEM data of the active time channel is drawn by small dots. The

noise level is shown by red horizontal dotted line(s) and the center time of the active channel is shown above the graph. The RMS error is shown at the left bottom corner of the graph.

As in the Data GUI, the *Channel #* text field and the *<-Chan* and *Chan->* push buttons are used to identify and change the active time channel. Pressing either button swaps to the previous or next time channel, but channels can be changed also by first writing the channel number in *Channel#* field and then pressing either of the two channel swapping buttons. *Swap component* push button will to swap the data component shown in the graph between Z and X components.

Show/hide measured button is used to hide the measured data from the results plot. *Show model* check box is used to replace the color scale of resistivity (or susceptibility) with an auxiliary graph showing the 2D resistivity model below the data profile (as in Fig. 2.9). Otherwise the graph shows the TEM response alone. The three check boxes *Use modified Log scale, Show all channels* and *Show hide grid* work like in the Data GUI, changing the appearance of the response graph. Also the *Zoom* and *Pan* buttons behave as in the Data GUI, allowing to see finer details of the fit.

Edit null weights and *Edit full weights* are buttons that give the user a possibility to define data weights for the inversion interactively. ArjunAir uses binary weights, which means that data points either included into or excluded from the inversion. The actual editing works as in data editing mode 1 (see chapter 2.2). The user draws a selection rectangle pressing the left mouse button and all the points inside the rectangle are then either removed from or added to the inversion depending on which button was pressed. After each selection the program asks the user whether the editing is made for all the time channels or only for the current one.

Pick sounding site button is used to interactively select a measurement site where sounding curve, i.e. a graph of the TEM response as a function of time, will be shown. The selection is made using mouse editing mode 2 (single left and right mouse click). A new GUI window appears showing a TEM sounding curve for the measurement site closest to the selected point. An example of the Sounding view is shown in Fig. 2.10. See chapter 2.8.1 for more information on the Sounding GUI window.

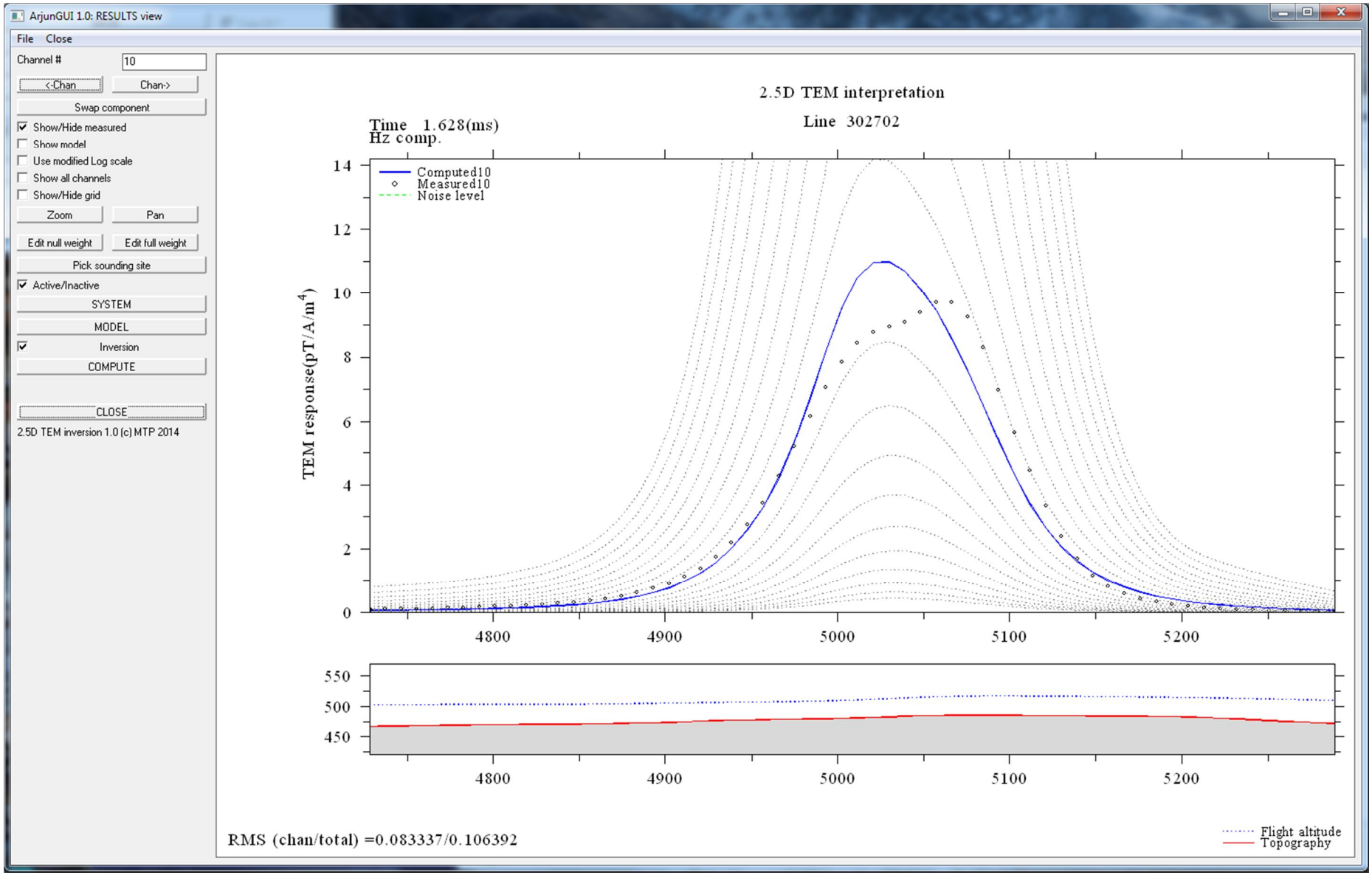


Figure 2.9 ArjunGUI Results view.

Active/Inactive button is used to inactivate (and consequently activate), the whole time channel (of current data component) from the inversion. Inactive time channel will not be included in the inversion and essentially it has the same meaning as if all points are given zero weight. Note that inactive channels and zero weighted data points are plotted differently on the Results GUI graph.

The remaining buttons (and *Inversion* check box) provide a quick access to the System GUI to change system parameters, Model GUI to edit the model, and Compute GUI to restart the inversion or forward computation.

Save TEM data and *Save XYZ data* menu items will save the computed and measured data into an AGD file and XYZ file, respectively. The remaining items in *File* menu allow saving the data graph in Postscript (PS), Encapsulated Postscript (EPS), Portable document format (PDF), Portable network graphics (PNG) and Graphic image format (GIF) files. The PDF file gives usually the best results and the snapshot tool of the PDF reader gives a possibility to create good quality bitmap image of the graphs.

Close menu contains two items. The first one is used to restart the Results GUI using either normal or modified aspect ratio. In normal mode 3:4 screen aspect ratio is used whereas in modified mode the program asks for screen ratio. A value less than one (default is 0.8) suits widescreen displays better because otherwise the GUI window would extend outside the right side of the screen. The second menu item will close the Model GUI and return the program operation back to Main or Results GUI.

2.8.1 TEM sounding GUI

The Sounding GUI window shows the TEM sounding curve of a single site as a function of time on a full logarithmic scale. An example is show in Fig. 2.10. The computed response is drawn with a solid blue curve and the measured data (if data has been red in) is shown by small open circles. The noise level is shown as a dotted horizontal green line. Data points whose absolute value is smaller than the noise level are marked by a black cross. The site number and the component name are shown above the graph and the site location (distance and XYZ coordinates) are shown below the graph.

The <-*Site* and *Site*-> push buttons are used to change the active time channel. Pressing either button swaps to the previous or next time site. Sites can be changed also by first writing the channel number in *Site#* field and then pressing either of the two site swapping buttons.

Swap component button changes the data component of the graph between Z and X components. *Show/Hide measured* check box hides the measured data from the graph.

Measured and computed data can be saved to a column formatted *.AGZ file using the *File/Save Sounding* menu item. Unlike the profile data in *.AGD file, sounding data are stored so that the columns refer to measurement sites and rows correspond to time channels.

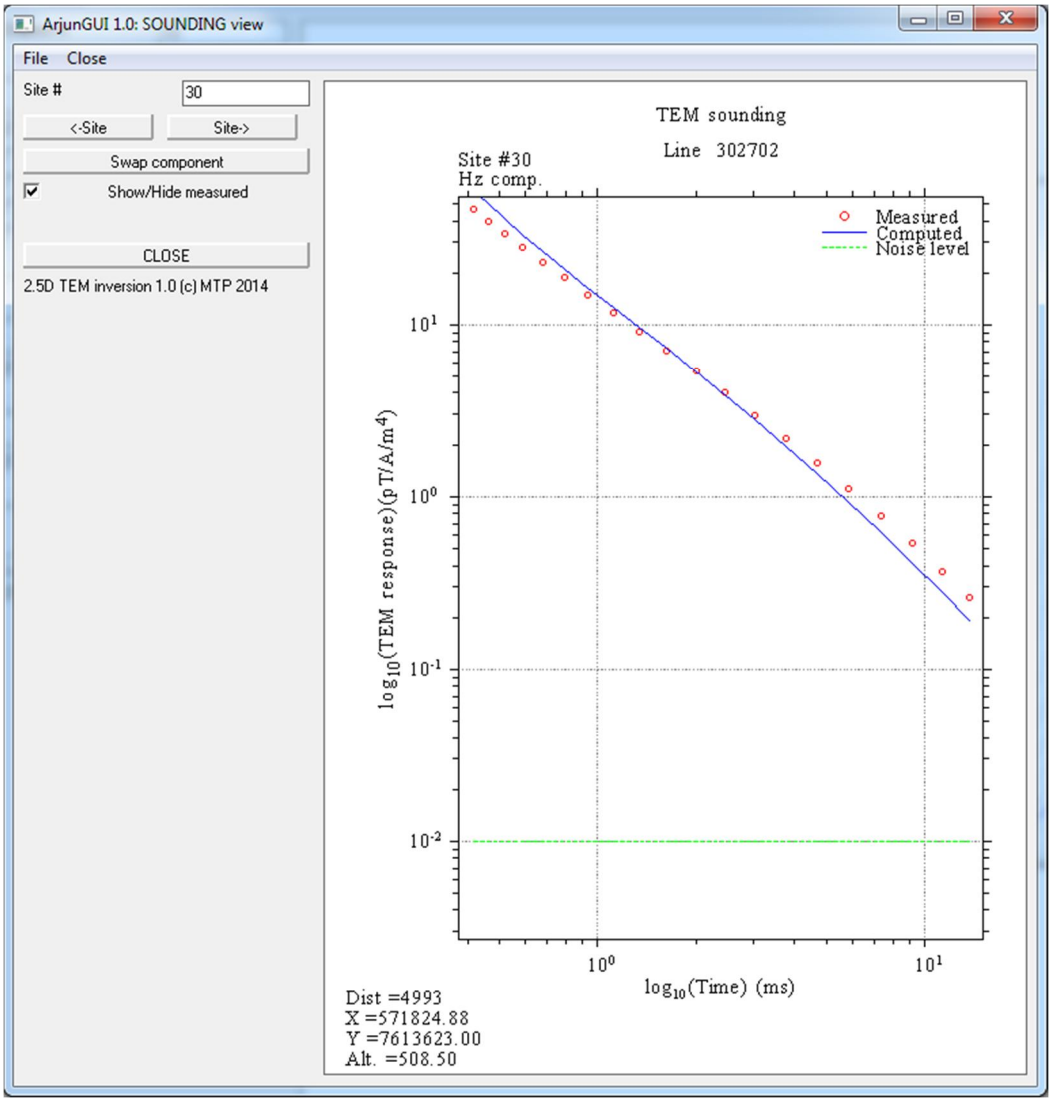


Figure 2.10 ArjunGUI Sounding view.

3. Program usage

The four main steps of TEM data inversion in practice are: 1) data input and preparation, 2) setting up computational parameters, 3) model creation and 4) forward computation or inversion. Guidelines to these tasks and other commonly encountered problems are discussed in the following.

3.1. Data input

Usually lots of airborne TEM data are measured at a time and all the data profile lines from the same measurement site are stored in single Geosoft XYZ file. The profile lines are tens of kilometers long and, thus, may contain anomalies of multiple conductors. ArjunGUI, however, works on a single profile and (preferably) a single anomaly at a time. Therefore, the first step is to read in a single profile from a XYZ file and select the part of the profile that contains the anomaly that requires interpretation. At this point it should be reminded that the inversion of a single anomaly is a time consuming process. High amplitude anomalies are usually caused by graphite bearing black schists and therefore their interpretation is seldom needed. In mineral exploration attention should preferably be paid on isolated anomalies of relatively low amplitude that display relatively high tau value ($\tau = (t_2\text{-}t_1)/(\log A_2\text{-}\log A_1)$).

After selecting (and possibly resampling) the interesting anomalous part of the profile one should save it as a new data file in *.AGD file format. When data are stored in AGD file format the user does not need to give any additional information when opening the same data again, which thus makes program operation faster. Moreover, because interpretation is a process that usually requires multiple steps the selected part of the data should be saved into a new folder the name of which should identify the line number and possible anomaly number or its coordinate along the profile (eg. LINE204_A and LINE204_B). Remember also that the AGD file preserves the original XY coordinates and the profile coordinate (distance from the beginning of the profile) of the original data.

3.2 Computational parameters

The system parameters, time channels and waveform parameters vary depending on the survey and the TEM field operator. Usually, however, they are the same for all the profiles of the survey site, and hence, their values need to be defined only once. And once proper parameters have been defined for a single survey, modifications to them are quite easy to make for each new case. In other words, use previous *.AGS, *.AGT and *.AGW files as a starting point for defining system, time channel and waveform parameters.

Apart from the most obvious system parameters (time domain data, db/dt vs. B-field, Z or X component etc.) the TEM data normalization is of great importance. Usually airborne TEM data is given in V/Am^4, which means that the measured voltage (V) has been divided first by the (effective) area of the receiver to get the dB/dt response and then it has been divided by the (effective) area of the transmitter to get response. Because ArjunAir computes the TEM dB/dt in T/s (and B-field in T), it is utterly important to check the *Normalize with NIA* check box at the center of the System GUI if the data is normalized with transmitter moment. Sometimes, the sign of the measured X component data is reversed and it may be needed to check also the *Reverse X component* check box too. Data normalization by primary field is usually used only for frequency-domain data that is not yet supported.

Regarding the transmitter and receiver parameters on the right side of the System GUI one should remember that in the actual computations (dipolar) point source and receiver are assumed. Nonetheless, the actual area and number of loop turns must be given for the receiver to be able to scale the data when *Normalize with NIA* option is used. At the moment, the offsets between the source and the receiver can be given separately for the Z and X components because that is the case in practice (eg. SkyTEM), but in ArjunAir computations they are the same – the offsets for the X receiver are the same as those of the Z receiver.

Definition of the time channels and the waveform is quite straightforward. The values of the current can be given as normalized, i.e., value of 1.0 is equal to the maximum transmitter current given in System GUI. The user should pay attention to the lower graph of the Waveform GUI and check that the zero time is given correctly so that the time channels are located inside the off-time range of the pulse. In addition, the value of the base frequency should comply with value provided by the TEM field operator. The number of time channels

defined in the *.AGP project file or *.AGT channel file can be different from the number of time channels in the *.AGD or *.XYZ files. In this case, the extra time channels are ignored.

3.3. Model setup

When new data is read in the existing model is not discarded by default. Usually, however, the old model and the new data do not match and, hence, the first thing to do is to press the *Discard model* button on Model GUI. Only after that a totally new initial model with correct X,Y and distance coordinates and default discretization can be computed pressing the *Generate initial model* button.

The minimum and maximum distance values of the initial model are computed automatically so that a cell node will locate below the first and last data point. The maximum depth of the model could be based on the maximum depth of exploration of the TEM system but in practice it can be given rather large value (eg. 600-1000 m) because the cell height is increasing downwards and usually one should be able to fix the bottom of the model to the background resistivity value.

The three last parameters on the input dialog of *Generate initial model* are the most important. The nominal horizontal node sampling is based on data intervals. To enable faster computation when building the first initial model it is quite OK to increase this value to make coarser grid.

Hint: To model the TEM response of the host medium accurately, the horizontal size of the elements should be less than the flight altitude.

To model the TEM response of an isolated target the horizontal size of the elements should be less than the point spacing.

Because the height of the cells is increasing monotonously, the default value of the nominal vertical node spacing (which is half of the horizontal spacing for the top row) is usually quite appropriate. The width of the margins however, should be such that the sides of the model are located sufficiently far away from the start and end of the profile. Usually the margins should be 200-400 m wide. Please, remember that the modified version of ArjunAir adds several

additional invisible rows and columns around the user defined mesh automatically to ensure that the edges of the model are so far (10-30 km) that they do not affect the computation.

If the data has a single narrow anomaly at the center of the profile the user might pay attention to the *Focus on center* check box and press the *Generate initial model* button again. In this case the horizontal discretization is doubled over the central part of the profile. Because the number of nodes increases by an amount of about 30%, it may be useful to make the horizontal discretization coarser when using the *Focus on center* option.

Initial resistivity model is also computed when *Generate initial model* is executed. The transformation, which is based on a special transformation of the TEM data, generates the shape of the initial resistivity distribution but may leave the resistivity values too large or too small in which case one should apply the *Mesh/Multiply values* menu item to scale the initial resistivity values according to the desired background resistivity value. The user has the possibility to start the inversion from a null-model in which case the user needs to reset the resistivity of the initial model using *Mesh/Reset resistivity* menu item. However, starting from a blank background model always require some additional iterations for the basic shape of the model to appear. Thus, the initial resistivity distribution can reduce the number of total iterations.

The fix/free status of the initial model is also set when *Generate initial model* is applied. The user should check the *Show fix/free* check box to verify that the resistivity distribution is confined mostly inside the area with free cells. Otherwise the user should either edit the fix/free status and make it wider or wipe the resistivity of the fixed cells. See chapter 2.6.1 for more information on interactive model editing. Note that the initial fix/free status of the cells depends greatly whether or not the *Focus on center* option is used.

Menu item *Define overburden* allows an easy way to add a conductive or resistive overburden layer on the top of the model. The overburden follows the topography of the surface and it contains a fixed number of cells.

Sometimes a more confined conductivity structure is desired, in which case the user needs to edit the resistivity values manually using the *Edit param.* or *Edit polygon* or *Paint points* options. Optionally the user also needs to edit the mesh geometry a using the *Double discr.* and *Half discr.* and *Shape cell* and *Stretch* options.

Unlike in frequency-domain measurements the magnetic susceptibility does not have as big role in time-domain measurements because the measurements are made off-time. An option to define the susceptibility model is still added in ArjunGUI to allow its use in FEM computation and possible joint inversion with static magnetic field data in the future.

3.4 Inversion and computation

Once all the prerequisites are set up and the *COMPUTE* button of the Main GUI is active, the user can start the forward computation or the inversion by checking the *Inversion* check box in the Main GUI.

If the data, model and computational parameters have not been changed since the program was run for the previous time, the user can apply the *Use previous results* check box in the Computation GUI. This makes the forward computation in a fraction of a time required for the full solution, because the ArjunAir.frq file contains the full frequency-domain solution to the modelling problem and the ArjunAir program only needs to perform the transformation to the time-domain. Thus, if the data file did not contain computed data it is possible to do the model computation very fast after opening the project file when program is restarted.

Hint: Whenever performing inversion in a sequence and the model or computational parameters are not changed manually after the last iteration, the user can apply the *Use previous results* option to bypass the full forward computation of the starting model. This almost halves the time needed for single inverse iteration.

For inversion, the program needs both ArjunAir.frq and ArjunAir.jcb file which contains the sensitivity matrix (Jacobian). Therefore, if inversion is to be made one iteration at a time, the first iteration is done normally after which the *Use previous results* option can be used between successive iterations.

The default mode of inversion is constrained Occam inversion with conjugate gradient solver (Occam+CG) using survey S-norm data normalization (per channel) and zero convergence (given number of iterations) is performed. The initial values of maximum parameter step (3 decades) and Lagrange scaler (L=1) are usually OK for the first few iterations. The Unconstrained (SVD) and Constrained (Occam+SVD) modes and the advanced options

(*Enhance roughness, Use depth weight and Subtract base level*) should be used for testing purposes only. The point P-norm can be used to fine tune the fit after the S-norm inversion has been used to find one possible solution.

The data are scaled so that the RMS error should converge towards zero when fit between the measured and computed data is very good. In practice, however, even 5% RMS error is very good. However, there is no way to tell how many iterations are required for good fit. Therefore, the *Target RMS percent* option should be used only when leaving the computer running on its own for a long time, for example overnight.

Hint: ArjunAir computation can be stopped by entering the Windows Task Manager (eg. pressing *Ctrl+Shift+Esc* keys), selecting the ArjunAir**.exe process and killing the application. ArjunGUI continues its operation as if computation has run into an error. Pressing *Ctrl+C* keys in ArjunGUI console window kills the whole application.

When the computation or inversion ends the Computation GUI closes automatically and the operation returns to the Main GUI or Results GUI. Although, the RMS error change shows up on the console window the user should always check the fit between measured and computed data by opening the Results GUI. The model shows up in Model GUI but it can also be added in the Results GUI applying the *Show model* check box. The user should apply the *<-Chan* and *Chan->* buttons to see the fit as a function of profile distance per each time channel. The user should also apply the *Pick sounding site* button to see the fit as a function of time.

Depending on the quality of fit and/or the resistivity model the user can adjust data weights to leave out some parts of the data (*Edit full weights* and *Edit null weights* buttons) or even whole time channels (*Active/Inactive* check box). Likewise the user can enter the Model GUI and edit the discretization, parameter values or fix/free status of the model so that the fit between measured and computed data gets smaller or a new alternative starting model is given for additional inverse iterations.

The interpretation of TEM data is a semi-interactive process that involves user discretion. Because of the non-uniqueness of the inverse EM problem, the model obtained from the inversion should not be treated as the only possible model. Instead, the user should try to find out which criteria are common among different models, what the maximum depth extent might be and how are the general size, shape and orientation of the target.

3.4.1 Forward modelling

The steps for forward modeling follow the abovementioned guidelines for data interpretation. Because in forward modelling data is not read in, there is no possibility to define variable flight altitude or topography. The Data GUI is only used to define the start and end coordinates of the profile, the constant flight altitude and the number of points on the profile (using the corresponding text fields that are inactive when data has been read in).

Once the system parameters, time channels and waveform have been defined the user should generate the initial model just like when data has been read in. The initial values of the model geometry are based on the profile length. The initial resistivity is equal to the user given background resistivity value. The user can add an overburden layer in the model before editing the resistivity (and susceptibility) model using the edit tools.

3.5 On the computational accuracy

For computational accuracy it is necessary to increase the discretization of high conductivity targets. During inversion the re-discretization of conductive targets should be done between the iterations. Unfortunately, there's no practical rule for the size of the cells vs. conductivity of the target. The rules based on skin depth ($\delta \approx \sqrt{2/\omega\sigma\mu}$), or its time-domain equivalent, the diffusion depth ($z = \sqrt{2t/\sigma\mu}$), usually lead to very fine discretization. In practice, the discretization is chosen as a compromise between the accuracy and computation time. In other words, the time that one wants to spend on the computation dictates how fine discretization is used for the most conductive parts of the model.

The computational accuracy of the original ArjunAir modelling code has not been tested thoroughly. Layered earth responses are correctly modelled provided that the mesh is extended far enough. Tests made using a thin conductive plate (by LeroiAir program) in the lower part of two-layer earth are modelled to an accuracy of about 20%. Despite the several improvements made to the original ArjunAir modelling code there still exists some computational problems. One of them is that the vertical component gets negative next to an outcropping contact with sufficiently high conductivity contrast. Simultaneous inversion of the vertical and horizontal component data does not work correctly either.

4. File formats

4.1 Data files (AGD)

AGD are column formatted text files that contain only one profile. Before trying to open an AGD data file, please, make sure that it's properly formatted. Two types of AGD files are supported: a) preformatted files and b) generic column formatted files. The header of a preformatted file contains all the necessary information required to read the data. When reading a generic column file, the header information must be provided interactively which is a slower process. Therefore, after opening a generic data file the user should to save the data in preformatted AGD file format.

The format of the preformatted input data file is shown below.

```
Skytem HM L 100901

15   80.00   1   0
 1   2   3   4  -5
 6  -2

 543060.6  7620940.0  2692.7  595.4  615.5  0.13381E+00  0.67038E-01 ...
 543051.0  7620940.0  2702.3  595.5  615.1  0.15396E+00  0.71438E-01 ...
 ...
```

- The 1.st line defines a header text (max. 80 characters), which is used as a secondary title above the response graph.
- The 3.rd line defines: 1) the number of time channels (NOC), 2) nominal flight altitude (ALT) used if altitude data is not available, 3) number of data components (NCOMP), and 4) on=1/off=0 flag for the existence of computed results. The maximum number of time channels is 50.
- The 4.th line defines the column indices of: 1 & 2) X and Y coordinates, 3) profile distance from the beginning of the profile, 4) Z coordinate i.e. the ground topography from sea level and 5) elevation or flight altitude. If the corresponding information is missing zero value should be used for the column index. If either X or Y coordinates are missing the data are put below the X axis. Both X and Y coordinates cannot be missing. If profile distance is missing it will be computed based on X and/or Y coordinates. If the column index of the altitude is negative the altitude is interpreted as elevation from sea level. Otherwise it is the altitude from ground level.

- The 5.th line defines the column indices of the NOC data channels of the first data component. If the second column index is negative it is interpreted as a column step so that, for example, every second column will be read.
- If second (horizontal, in-line) data component were to be read, the 6.th line would define its column indices (just like the columns if he first data component).
- If the computed data were to be read from the file, the 6.th line (or 8.th line if second data component is to be read) would define its column indices.
- In this case, the data columns start from the 7.th line.

All data columns must be real (floating point) values. Thus, AGD files cannot contain commonly used date and time information, containing slash "/" and colon ":" characters. Lines starting with hatch "#" or slash "/" characters or with an exclamation mark "!" are treated as comment lines.

4.1.2 Generic data files

If the first character on the first line of the input data file is a comment character ("#", "/" or "!") the file is assumed to be a generic (single profile) data file. In that case the program interactively asks the user for the necessary (header) information to read the correct columns from the file. The remaining lines must contain the (real valued) data columns. Empty lines and comment lines are ignored. To be able to read data file with a suffix other than AGD one needs to change the file type mask at the bottom of the Open/Save dialog.

4.2 Geosoft XYZ files

Geosoft XYZ files are column formatted text files that can contain multiple measurement lines or profiles lines. ArjunGUI supports XYZ files only partially. This means that although ArjunGUI can read data from XYZ files (*File/Read XYZ file* menu item in Data GUI), the data should be saved in AGD file format so that it can be read in automatically the next time the program is started. An example of an XYZ file is given below:

```
/ ------------------------------------------------------------------ ...
/ XYZ EXPORT [05/15/2013]
/ DATABASE   [.\SkyTEM_LMHM_Selected_EM_MAG.gdb]
/ ------------------------------------------------------------------ ...
/         Fid       Line        Flight           DateTime       Date ...
/============ ========== ============= ================== ========== ...
//Flight 90302
//Date 2012/09/03
Line 100901
  1089448.30     100901   20120903.02   41155.609 2012/09/03 ...
  1089448.40     100901   20120903.02   41155.609 2012/09/03 ...
  1089448.50     100901   20120903.02   41155.609 2012/09/03 ...
  ...
```

Lines beginning with a slash character `"/"` are comment lines. The beginning of each line is marked by directive `"LINE"` followed by an identifier that is usually the line number but can also be a character string. XYZ files do not require that the columns are labelled but this is highly advantageous, because usually TEM data files contains tens of columns and Read XYZ GUI can be used to identify and assign the column indices quite easily. If column labels are used they should be added to the file header before the first instance of the LINE directive. See chapter 2.1.2 for more details about reading XYZ files.

4.3 System parameters files (AGS)

The system parameters are stored in *.AGS files as shown below:

```
#ArjunGUI: System parameters

  1   1   1   1   2   1
  1       10.00   100000.00
  8        0.00      529.00
     2.00   16.70     0.00    2.00   16.70  0.00
     116.00   1.00    1.00
  1   1
  0.001000   0.001000
```

- The first line is just a header defining the contents of the file.
- The 3rd line defines:
  1) domain mode (DMODE= 1 and 2 for automatic and user defined TEM computation and DMODE= 3 for FEM computations),
  2) response mode (RMODE= 1 for dB/dt and RMODE= 2 for B-field),
  3) component mode (CMODE= 1 for Z (vertical) component alone, CMODE= 2 for X (in-line) component and CMODE= 3 for both Z and X component data,

4) normalization mode (NMODE= 1 for un-normalized data, NMODE= 2 for data normalized by vertical primary field, NMODE= 3 for horizontal primary field normalization, NMODE= 4 for components's own total field , and NMODE= 5 for data normalized with the total intensity of the (dB/dt or B) field.
5) data unit mode (TUNIT= 1 for nano, TUNIT= 2 for pico, TUNIT= 3 for femto T/s
6) normalized data unit mode (NUNIT= 1 for percent (%), NUNIT= 2 for permille (), NUNIT= 3 for permillion (ppm), NUNIT= 4 for perbillion (ppb).

- The 4.th line defines: 1) points-per-decade mode (PPD), and 2 & 3) the minimum and maximum frequency (MINF, MAXF) for the user defined TEM computation mode (DMODE= 2). See chapter 2.3 for more information on PPD modes.
- The 5.th line defines: 1) the number of turns in the transmitter loop, 2) the angle of the transmitter loop (degrees from vertical, positive if the front of the loop is above the rear of the loop), and 3) the surface area (m^2) of the transmitter loop.
- The 6.th line defines: the Z, X and Y offsets (m) between the source and Z component receiver and the X component receiver. The values are positive below, behind and left to the transmitter loop, respectively.
- The 7.th line defines: 1) the current (A) used in data normalization and 2 & 3) the effective surface area (m^2) of the Z and X component receivers. Presently, the receiver surface areas are not used in ArjunGUI.
- The 8.th line defines: 1) on=1/off=0 flag for TEM response normalization, 2) on=1/off=0 flag for the reversal the sign of the X component data. TEM response normalization is needed when measured data are defined as $V/(Am^4)$ instead of V/m^2 = T/s for the dB/dt response (RMODE= 1). Response normalization means that in the inversion the measured data is multiplied with transmitter moment (NIA) so that ArjunAir can minimize the difference between response defined as T/s. For data display, the computed response is then divided by NIA so that the response is $V/(Am^4)$.
- The 9.th line defines the noise level of the Z and X component data. All data whose absolute value is below the noise level are given zero weight in the inversion.

4.4 Time channel files (AGT)

The time channels are stored in *.AGT files as shown below:

```
#ArjunGUI: Time_channels

 1  15   15.000   5.000

#Start_ms, endtime_ms
   0.3044      0.3830
   0.3834      0.4830
   0.4834      0.6090
    ...
```

- The first line is just a header defining the contents of the file.
- The 3rd line defines: 1) the time channel mode (TMODE= 1/2), 2) number of time channels (NOT), 3) the length of the off-time period (ms), and 4) the location of zero-time (ms) in the waveform. See TMODE options below.
- From the 6.th line onwards are pairs of either a) the start and end times of each time channel (TMODE= 1) or b) the center and width of each time channel (TMODE= 2). Normally, time values are defined in milliseconds. However, if time channel mode is negative (TMODE= -1 or -2), time values are assumed to be in seconds and will be converted appropriately. Alternatively, if TMODE= 3 or 4 the time channel values are assumed to be in microseconds. Note that off-time and zero-time must still be defined in milliseconds.

4.5 Waveform files (AGW)

The waveform is stored in *.AGW files as shown below:

```
#ArjunGUI: Waveform

 1   1   0  4   15.000   5.000

#Time_ms, current_A
  0.0000     0.0000
  0.5000   116.0000
  4.8000   116.0000
  5.0000     0.0000
```

- The first line is just a header defining the contents of the file.
- The 3rd line defines: 1) waveform mode (WMODE= 1/2), 2) calibration data mode (XMODE), 3) number of points that define the waveform, 4) the length of the off-time period (ms), and 5) the location of zero-time (ms) in the waveform. If WMODE= 1 waveform is defined by the current (A) otherwise waveform is based on calibration data.

The calibration data is either a) in-line dB/dt (XMODE= 1), b) in-line B-field (XMODE= 2), c) vertical dB/dt (XMODE= 3) or d) vertical B-field (XMODE= 4).

- From the 6.th line onwards are pairs of either a) the time and current values or b) the time and calibration data values. Normally, time values are given in milliseconds but if waveform mode is negative (WMODE= -1 or -2) the values are assumed to be given in seconds and will be converted to milliseconds. The scaling, unit and normalization of the calibration data is defined by system parameters (nano, pico or femto, T/s=V/m^2 or T, V/(Am^4) or T/(Am^2).

4.6 Model files (AGM)

An example of the 2D resistivity and susceptibility model file (*.AGM) is given below:

```
#Model for ArjunGUI  1.0

  102   29    9
  4710.00    5304.00    485.60   -140.80
  600.00       9.00      4.00   1000.00
  0.001000  10000.0   0.000010     10.0
  1  1  2  0  0  0   0.500   1.500   0.100

  4521.00    467.20  0.10000E+04  0.00000E+00  1
  4521.00    463.20  0.10000E+04  0.00000E+00  1
  ...
```

- The first line is just a header defining the contents of the file.
- The 3rd line defines: 1 & 2) the number of mesh nodes in X and Z direction (INODE & JNODE) and 3) the number of nodes used for the margins before and after the beginning and end of on the profile (IMARG)
- The 4.th line defines: 1 & 2) the beginning and end X coordinate of the model (m), 3 & 4) the top and bottom Z coordinates of the model (m). The top of the model is equal to the highest topographic value in the mesh. Like the left and right side of the model, the bottom is always flat.
- The 5.th line defines: 1) maximum depth (m), 2 & 3) X and Z distance step (m) and 4) background resistivity value (Ωm) used to create the initial model.
- The 6.th line defines: 1 & 2) minimum and maximum resistivity value, 3 & 4) minimum and maximum value of magnetic susceptibility. These values are used for the color scale in Model GUI and as limit values in the resistivity inversion.

- The 7.th line defines various inversion parameters that were (possibly) used when the model was obtained by inversion. The parameters are: 1) inversion norm (survey vs. point), 2) convergence criterion (zero level vs. target RMS), 3) inversion method (SVD vs. Occam+CG vs. Occam+SVD), 4) removal of base level (on/off), 5) enhanced roughness (on/off), 6) depth weighting (on/off), 7) Lagrange multiplier, 8) maximum parameter ($\log_{10}$) step, and 9) weight for X component data.
- From the 9.th line onwards the (TNODE= INODE*JNODE) rows are the: 1 & 2) X and Z coordinates of the nodes (m), 3 & 4) resistivity (Ωm) and susceptibility (dimensionless, SI) of the element between (INODE,INODE+1) and (JNODE,JNODE+1), and 5) the fix/free status of the element (1= fixed, 0= free). The background resistivity and susceptibility values are used for the elements on the right side and on the bottom of the model.

4.7 Output mesh files (DAT)

In model file (*.AGM) the X coordinates of the nodes are based on the distance from the beginning of the original (un-zoomed) measurement line. The actual geographic X and Y coordinates are not saved in the model file. To create a general purpose file that can be used to visualize the model in CAD and GIS software, the model can be saved as a column formatted text file (*.DAT), that contains the X, Y, Z coordinates, the distance values, the resistivity and susceptibility values and fix/free status of the cells below the profile (i.e. without the margins). Note that the number of cells in the whole mesh is (NCELL= (INODE-1)*(JNODE-1)) and since the margin areas are not saved the number of cells in output file is NCELL-2*IMARG*(JNODE-1).

Mesh can be saved using menu item *File/Save mesh* DAT in MODEL view. The example below shows the format of mesh file output:

```
#Mesh from ArjunGUI
#x, y, d, z, res_ohmm, sus_SI

 538510.81  7620794.00  3626.500   560.352  0.10000E+05  0.00000E+00  1
 538476.31  7620486.00  3637.500   561.190  0.10000E+05  0.00000E+00  1
 538472.56  7620636.50  3648.500   562.193  0.10000E+05  0.00000E+00  1
 ...
```

4.8 GUI parameter file (DIS)

Several GUI related parameters are automatically written in and read from the ArjunGUI.dis file. This allows translating some of the graphical text elements into other languages if needed. Note that the format of the *.DIS file must be preserved. If the format becomes invalid (an error message appears) you should delete the ArjunGUI.dis file and a new one with default values will be generated automatically the next time the program is started.

The file format and default values are shown below.

```
# ArjunGUI 1.00 parameter file

   26   24   22   18
  400   400       0.85       0.85       0.80
  1  2  1  1  1  0
    0.000010    100000.0   0.0000100      10.000
 12 13 18 19 43  1 45  1

2.5D TEM interpretation
TEM data
2D model
Distance (m)
TEM response
Channel
Time
Z & H (m)
Topography
Elevation
Noise level
Time (ms)
Distance (m)
Depth (m)
TEM data
Resistivity ({M2}W{M1}m)
Susceptibility (SI)
Waveform
Current (A)
Calibr. (nT/s)
Calibr. (nT)
Channels
Zerotime
Computed
Measured
Bx
Bz
dBx/dt
dBz/dt
```

- The 1.st line is a comment line defining the program and file version
- The 3.rd line defines four different character heights used, for example, for the main title and axis labels, plot legends, and resistivity labels in the vertical cross-sections.

- The 4.th line defines: 1) the x (horizontal) and y (vertical) position of the origin of the graph (in pixels) from the bottom-left corner of the page, 2 & 3) the length of the x and y axes relative to the width and height of the page, respectively, and 4) the aspect ratio for widescreen mode. Note that the origin is defined as an integer value relative to the page size which is (in printable graphs) 2970 x 2100 pixels corresponding to A4 paper size.
- The 5.th line has five integer valued parameters that define: 1) normal/widescreen mode, 2) color scale index, 3) mesh outline mode, 4) show/hide status of the grid, 5) show/hide status of the flight line, and 6) ArjunAir**.exe version flag: 0= 32 bit & 1=64bit.
- The 6.th line defines: 1-4) the minimum maximum values of the resistivity and susceptibility, and 5) effective radius of mesh stretching and resistivity painting.
- The 7.th line defines the column indices used for the previous XYZ file.
- Lines after the 8.th line contain various text items of the graphs (max. 40 characters). These include the main title, axis titles and legend labels of various graphs.

Special characters "{" and "}" should not be used in any text strings, because they define instruction strings that enable Greek symbols (see Ωm in the example above). Other instruction characters that must be avoided are: "^" for superscripts, "~" for subscripts and "_" for setting the baseline back to its original position. Please, see DISLIN manual for more information about the topic.

4.1 Sounding files (AGZ)

AGZ files are column formatted text files that contain the data in sounding format. That means the columns contain the data for each site (receiver) and the rows correspond to the time channels. They can be saved only from within the Sounding GUI. An example is shown below.

```
#TEM sounding data saved by ArjunGUI
  time(ms)     1mz         1cz         2mz         2cz         3mz
 0.41770 0.59579E+00 0.76389E+00 0.70361E+00 0.93195E+00 0.85946E+00 ...
 0.46170 0.46757E+00 0.60586E+00 0.55812E+00 0.74026E+00 0.68881E+00 ...
 0.51720 0.36447E+00 0.47164E+00 0.44094E+00 0.57599E+00 0.55057E+00 ...
 ...
```

The first line is a comment line defining the contents of the file. The second line contains column labels. The first column defines the center time (ms) of the time channels. The next

two columns contain the measured and computed data for the first site (1mz & 1cz), then comes the second site (2mz, 2cz), etc. If x component is available it is saved after the last column of the z component data. If either measured or computed data is missing then the corresponding columns will be missing too.

5. Additional information

ArjunGUI was developed in 2012-2014 at the University of Oulu by Dr. Markku Pirttijärvi in NovTecEx project funded by TEKES (Finnish Funding Agency for Innovation) Green Mining programme. The project was hosted by Geological Survey of Finland and supported by Agnico Eagle Mines (Kittilä mine), Anglo American Exploration (Sakatti Mining), First Quantum Minerals (Kevitsa mine), Inmet Mining (Pyhäsalmi mine), and Metsähallitus (Administration of Forests).

ArjunGUI uses a modified version of ArjunAir developed by Glenn Wilson, Art Raiche and Fred Sugeng for the CSIRO/Amira consortia in project P223F (Wilson et al., 2006). The original source code in Fortran language (version 7.0.5) is available at http://p223suite.sourceforge.net/.

The main modifications are as follows: 1) a global module is used to pass large allocatable arrays between subroutines, 2) unnecessary temporary file I/O is replaced with global arrays, 3) sensitivity matrix (Jacobian) is trimmed by removing fixed resistivity cells and zero-weighted data, 4) original SVD inversion method was replaced with: a) unconstrained SVD based inversion with adaptive damping, b) constrained (Occam) inversion with (faster) iterative conjugate gradient solver, and c) constrained Occam inversion using (slower) SVD solver, 5) wider and more carefully positioned mesh is extended on all four sides of the user defined mesh, 6) old results can be used also in inversion, because both the frequency domain spectrum (*.FRQ) and the Jacobian (*.JCB) are saved after inversion, 7) OpenMP parallelisation are used in the 64-bit version of ArjunAir (requires the libiomp5md.dll file).

Several subroutines including the SVD algorithm and the conjugate gradient solver have been adapted from Press et al. (1988). The SVD suits both under- and over-determined problems. The unconstrained SVD parameter optimization is based on a linearized inversion method, where singular value decomposition (SVD) with adaptive damping is used. The inversion

method has been described in my PhD thesis (Pirttijärvi, 2003). The constrained Occam inversion algorithm is derived from Grablox2 – gravity interpretation and modelling software using a 3D block model (Pirttijärvi, 2011).

ArjunGUI was written in Fortran-90 style and compiled with Intel Fortran 14 (Intel Visual Fortran Composer XE). The graphical user interface is based on the DISLIN graphics library (version 10.2) by Helmut Michels (http://www.dislin.de). Since DISLIN graphics library is available for other operating systems (Linux, Mac) ArjunGUI could be compiled and run on other operating systems without major modifications. However, the source code will not be made available in the near future. If you find the computed results erroneous or have suggestions for improvements, please, inform me.

5.3 Terms of use and disclaimer

ArjunGUI version 1.0 is freeware. It must not be distributed commercially.

If you publish the results computed with ArjunGUI, please, provide a notice "ArjunGUI © University of Oulu" (without the quotes). As a reference, please, use this user's manual "Pirttijärvi, 2014. ArjunGUI – 2.5D modelling and inversion of time-domain EM data; User's guide to version 1.0, University of Oulu".

The program is provided as is. The author (M.P.) and the University of Oulu disclaim all warranties, either expressed or implied, with regard to this software. In no event shall the author or the University of Oulu be liable for any indirect or consequential damages or any damages whatsoever resulting from loss of use, data or profits, arising out of or in connection with the use or performance of this software. All in all, use ArjunGUI at your own risk.

References

Pirttijärvi M., 2003. Numerical modeling and inversion of geophysical electromagnetic measurements using a thin plate model. PhD thesis, Acta Univ. Oul. A403, Univ. of Oulu < http://urn.fi/urn:isbn:951427119X>

Pirttijärvi M., 2011 Grablox2 - Gravity interpretation and modeling software based on 3-D block models; User's guide to version 2.0. University of Oulu, Department of Physics <https://wiki.oulu.fi/x/jYU7AQ>

Press, W.H., Flannery, B.P., Teukolsky, S.A. and Vetterling, W.T., 1988. Numerical Recipes, The art of scientific computing. Cambridge University Press.

Wilson, G.A., Raiche, A.P. & Sugeng, F., 2006. 2.5D inversion of airborne electromagnetic data. Expl. *G*eoph., 37, 363-371.